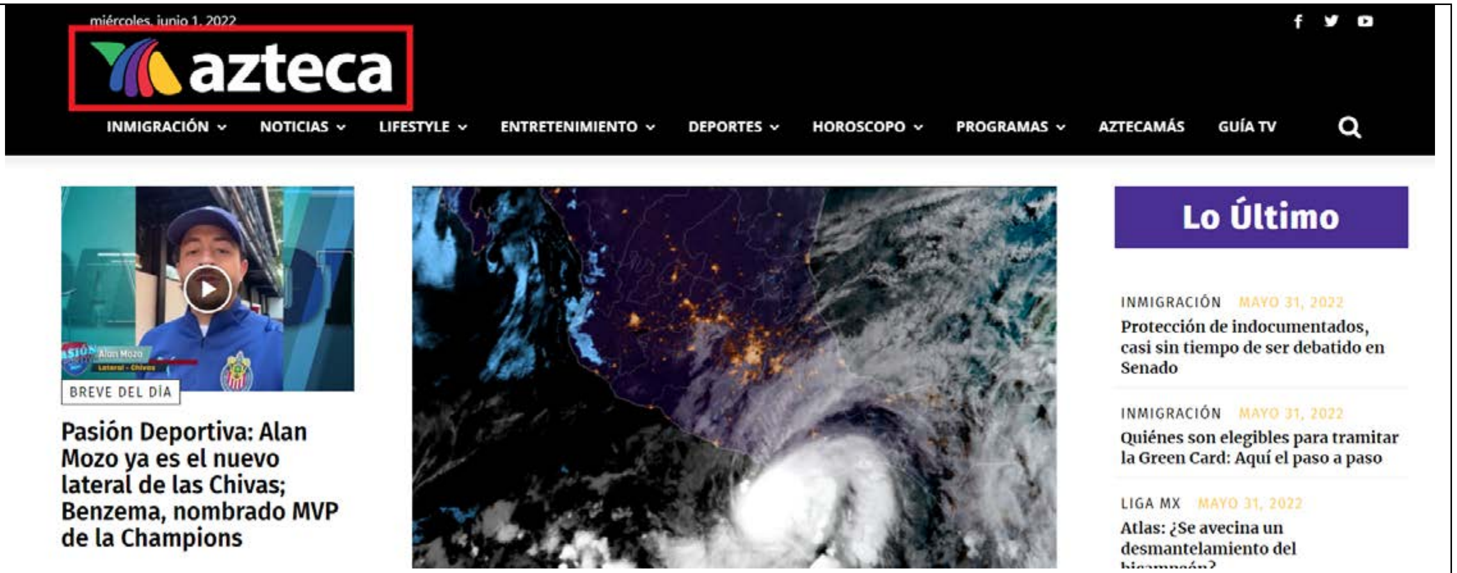
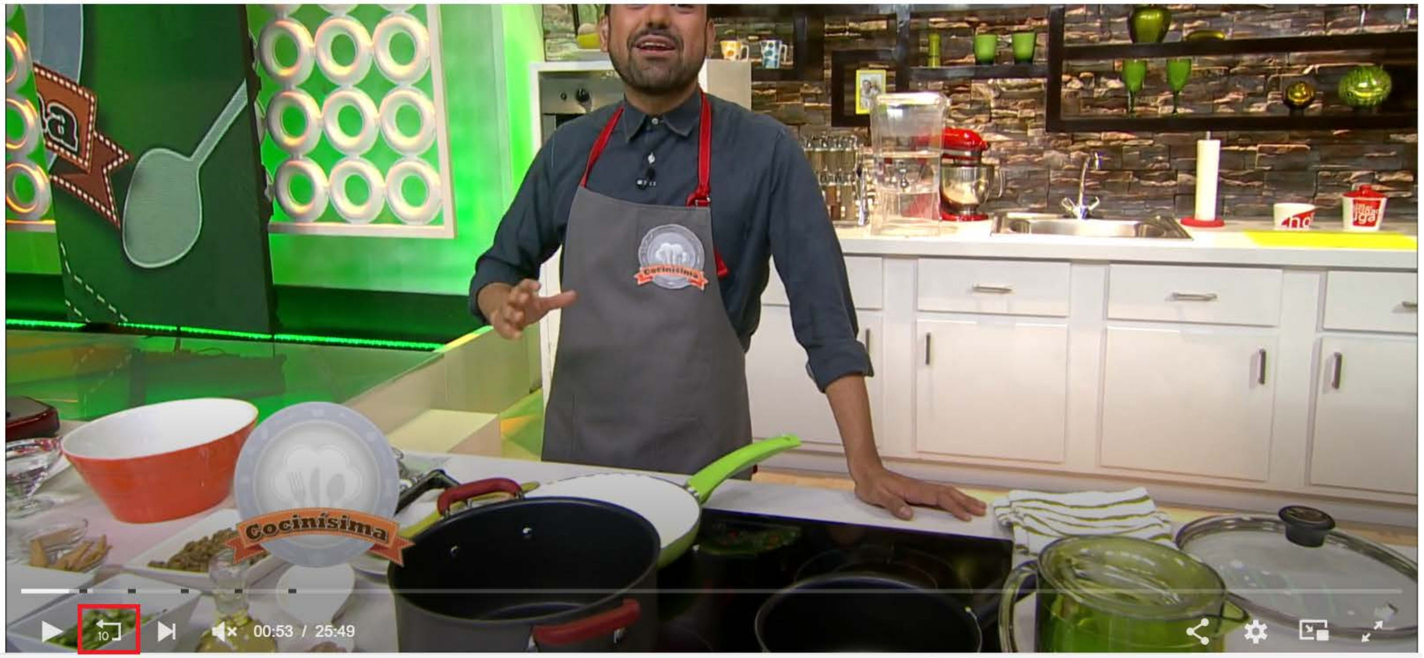
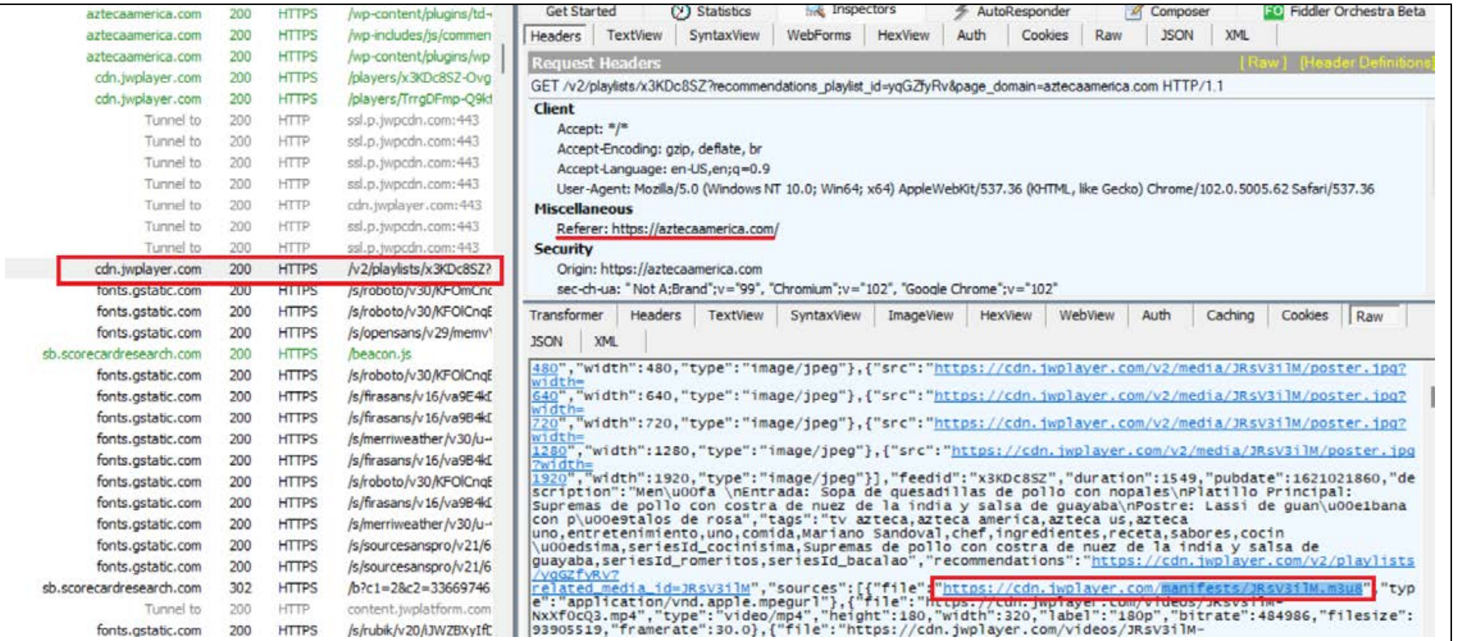
EXHIBIT F

US8195828B2	Azteca America
1. A method for discontinuous transmission, in sections, of encoded video data in a network of distributed appliances, the method comprising the following steps:	Azteca America supports HLS streaming protocol (“the Standard”). In addition, Azteca America utilizes HLS for delivery of contents to its customers/viewers. As shown below, a video content from Azteca America is streamed and the data traffic is captured showing the media format as HLS, the m3u8 file, (e.g., the Media playlist file comprising links to content chunks in .ts format used by HLS to contain information about the media playing), and the encryption scheme used by the streamed video. In addition, the HLS stream provided through Azteca America provides trick mode operation (e.g., 10 sec. reverse playback, etc.) to the streamed video. On information and belief, Defendant performs all steps of this claim or, alternatively, to the extent a user performs any step, Defendant conditions the user’s use of the Defendant’s accused instrumentalities using the Standard on the performance of that step as disclosed herein. For example, on information and belief, a user cannot use the accused instrumentality utilizing the Standard as described in this claim chart without performance of the steps recited in this claim. By providing the accused instrumentality utilizing the Standard as disclosed herein, Defendant also controls the manner and/or timing of the functionality described in this claim chart. In other words, for a user to utilize the functionality described in this claim chart, the steps of this claim must be performed in the manner described herein. Without performance of the steps as described herein, the Defendant’s functionality will not be available to users. The HLS standard practices a method for discontinuous transmission (e.g., discontinues transmission of video data during a trick mode such as fast-forward, fast-rewind, etc. in which an I-frame only version is transmitted), in sections (e.g., media segments), of encoded video data (e.g., Codec encoded representations of video stream) in a network of distributed appliances (e.g., network of HLS distribution servers and client devices).

	 miércoles, junio 1, 2022 azteca INMIGRACIÓN NOTICIAS LIFESTYLE ENTRETENIMIENTO DEPORTES HOROSCOPO PROGRAMAS AZTECAMÁS GUÍA TV BREVE DEL DÍA **Pasión Deportiva: Alan Mozo ya es el nuevo lateral de las Chivas; Benzema, nombrado MVP de la Champions** **Lo Último** INMIGRACIÓN MAYO 31, 2022 Protección de indocumentados, casi sin tiempo de ser debatido en Senado INMIGRACIÓN MAYO 31, 2022 Quiénes son elegibles para tramitar la Green Card: Aquí el paso a paso LIGA MX MAYO 31, 2022 Atlas: ¿Se avecina un desmantelamiento del https://aztecaamerica.com/

	 https://aztecaamerica.com/ Shown below is the URL of .m3u8 master file sent by Azteca America server which identifies the usage of HLS based streaming by Azteca America servers. The m3u8 master file refers to all the variants of the video encoded for various bandwidths and resolutions. The URL of .m3u8 master file is: https://cdn.jwplayer.com/manifests/JRsV3ilM.m3u8

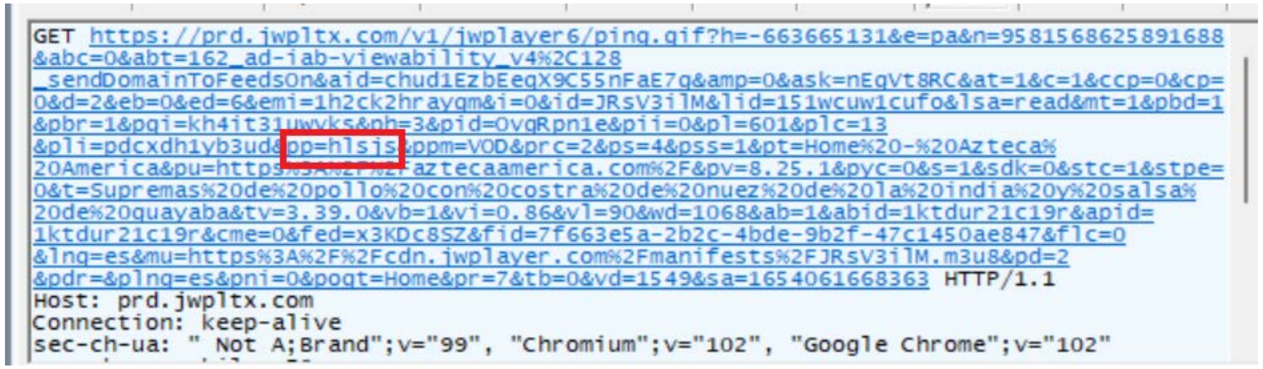


```
aztecaamerica.com             200  HTTPS  /wp-content/plugins/td-
aztecaamerica.com             200  HTTPS  /wp-includes/js/commen
aztecaamerica.com             200  HTTPS  /wp-content/plugins/wp
cdn.jwplayer.com              200  HTTPS  /players/x3KDc8SZ-Ovg
cdn.jwplayer.com              200  HTTPS  /players/TrrgDFmp-Q9kt
Tunnel to                     200  HTTP   ssl.p.jwpcdn.com:443
Tunnel to                     200  HTTP   ssl.p.jwpcdn.com:443
Tunnel to                     200  HTTP   ssl.p.jwpcdn.com:443
Tunnel to                     200  HTTP   ssl.p.jwpcdn.com:443
Tunnel to                     200  HTTP   cdn.jwplayer.com:443
Tunnel to                     200  HTTP   ssl.p.jwpcdn.com:443
Tunnel to                     200  HTTP   ssl.p.jwpcdn.com:443
cdn.jwplayer.com              200  HTTPS  /v2/playlists/x3KDc8SZ?
fonts.gstatic.com             200  HTTPS  /s/roboto/v30/KFOmCnc
fonts.gstatic.com             200  HTTPS  /s/roboto/v30/KFOlCnqE
fonts.gstatic.com             200  HTTPS  /s/opensans/v29/memv
sb.scorecardresearch.com      200  HTTPS  /beacon.js
fonts.gstatic.com             200  HTTPS  /s/roboto/v30/KFOlCnqE
fonts.gstatic.com             200  HTTPS  /s/firasans/v16/va9E4kD
fonts.gstatic.com             200  HTTPS  /s/firasans/v16/va9B4kD
fonts.gstatic.com             200  HTTPS  /s/merriweather/v30/u-
fonts.gstatic.com             200  HTTPS  /s/firasans/v16/va9B4kD
fonts.gstatic.com             200  HTTPS  /s/roboto/v30/KFOlCnqE
fonts.gstatic.com             200  HTTPS  /s/firasans/v16/va9B4kD
fonts.gstatic.com             200  HTTPS  /s/merriweather/v30/u-
fonts.gstatic.com             200  HTTPS  /s/sourcesanspro/v21/6
fonts.gstatic.com             200  HTTPS  /s/sourcesanspro/v21/6
sb.scorecardresearch.com      302  HTTPS  /b?c1=2&c2=33669746
Tunnel to                     200  HTTP   content.jwplatform.com
fonts.gstatic.com             200  HTTPS  /s/rubik/v20/iJWZBXyIfD
```

```
Request Headers
GET /v2/playlists/x3KDc8SZ?recommendations_playlist_id=yqGZfyRv&page_domain=aztecaamerica.com HTTP/1.1
Client
    Accept: */*
    Accept-Encoding: gzip, deflate, br
    Accept-Language: en-US,en;q=0.9
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.5005.62 Safari/537.36
Miscellaneous
    Referer: https://aztecaamerica.com/
Security
    Origin: https://aztecaamerica.com
    sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="102", "Google Chrome";v="102"
```

```
480","width":480,"type":"image/jpeg"},{"src":"https://cdn.jwplayer.com/v2/media/JRsV3ilM/poster.jpg?
width=
640","width":640,"type":"image/jpeg"},{"src":"https://cdn.jwplayer.com/v2/media/JRsV3ilM/poster.jpg?
width=
720","width":720,"type":"image/jpeg"},{"src":"https://cdn.jwplayer.com/v2/media/JRsV3ilM/poster.jpg?
width=
1280","width":1280,"type":"image/jpeg"},{"src":"https://cdn.jwplayer.com/v2/media/JRsV3ilM/poster.jpg
?width=
1920","width":1920,"type":"image/jpeg"}],"feedid":"x3KDc8SZ","duration":1549,"pubdate":1621021860,"de
scription":"Men\u00fa \nEntrada: Sopa de quesadillas de pollo con nopales\nPlatillo Principal:
Supremas de pollo con costra de nuez de la india y salsa de guayaba\nPostre: Lassi de guan\u00e1bana
con p\u00e9talos de rosa","tags":"tv azteca,azteca america,azteca us,azteca
uno,entretenimiento,uno,comida,Mariano Sandoval,chef,ingredientes,receta,sabores,cocin
\u00edsima,seriesId_cocinisima,Supremas de pollo con costra de nuez de la india y salsa de
guayaba,seriesId_romeritos,seriesId_bacalao","recommendations":"https://cdn.jwplayer.com/v2/playlists
/yqGZfyRv?
related_media_id=JRsV3ilM","sources":[{"file":"https://cdn.jwplayer.com/manifests/JRsV3ilM.m3u8","typ
e":"application/vnd.apple.mpegurl"},{"file":"https://cdn.jwplayer.com/videos/JRsV3ilM-
NxXf0cQ3.mp4","type":"video/mp4","height":180,"width":320,"label":"180p","bitrate":484986,"filesize":
93905519,"framerate":30.0},{"file":"https://cdn.jwplayer.com/videos/JRsV3ilM-
```

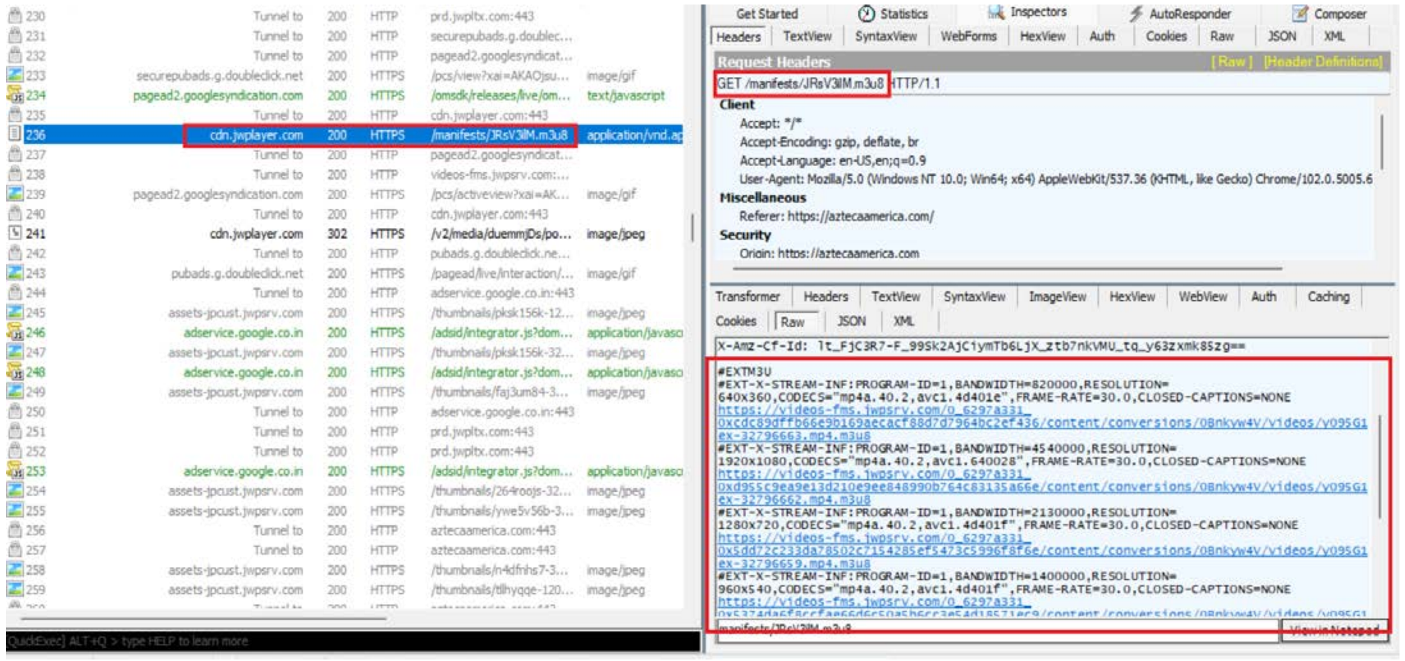
Source: Packet capturing by fiddler tool



```
GET https://prd.jwpltx.com/v1/jwplayer6/ping.gif?h=-663665131&e=pa&n=958156862589168
&abc=0&abt=162_ad-iab-viewability_v4%2C128
_sendDomainToFeedsOn&aid=chud1EzbEeqX9C55nFaE7g&amp=0&ask=nEgVt8RC&at=1&c=1&ccp=0&cp=
0&d=2&eb=0&ed=6&emi=1h2ck2hraygm&i=0&id=JRsV3ilM&lid=151wcuw1cufo&lsa=read&mt=1&pbd=1
&pbr=1&pgi=kh4it31uwyks&ph=3&pid=OvqRpn1e&pii=0&pl=601&plc=13
&pli=pdcxdh1yb3ud&pp=hlsjs&ppm=VOD&prc=2&ps=4&pss=1&pt=Home%20-%20Azteca%
20America&pu=https%3A%2F%2Faztecaamerica.com%2F&pv=8.25.1&pyc=0&s=1&sdk=0&stc=1&stpe=
0&t=Supremas%20de%20pollo%20con%20costra%20de%20nuez%20de%20la%20india%20y%20salsa%
20de%20guayaba&tv=3.39.0&vb=1&vi=0.86&vl=90&wd=1068&ab=1&abid=1ktdur21c19r&apid=
1ktdur21c19r&cme=0&fed=x3KDc8SZ&fid=7f663e5a-2b2c-4bde-9b2f-47c1450ae847&flc=0
&lng=es&mu=https%3A%2F%2Fcdn.jwplayer.com%2Fmanifests%2FJRsV3ilM.m3u8&pd=2
&pdr=&plng=es&pni=0&pogt=Home&pr=7&tb=0&vd=1549&sa=1654061668363 HTTP/1.1
Host: prd.jwpltx.com
Connection: keep-alive
sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="102", "Google Chrome";v="102"
```

Source: Packet capturing by fiddler tool

The Azteca America server sends to its users/subscribers references of all the variants of the stream (based on resolution and bandwidth) as follows.



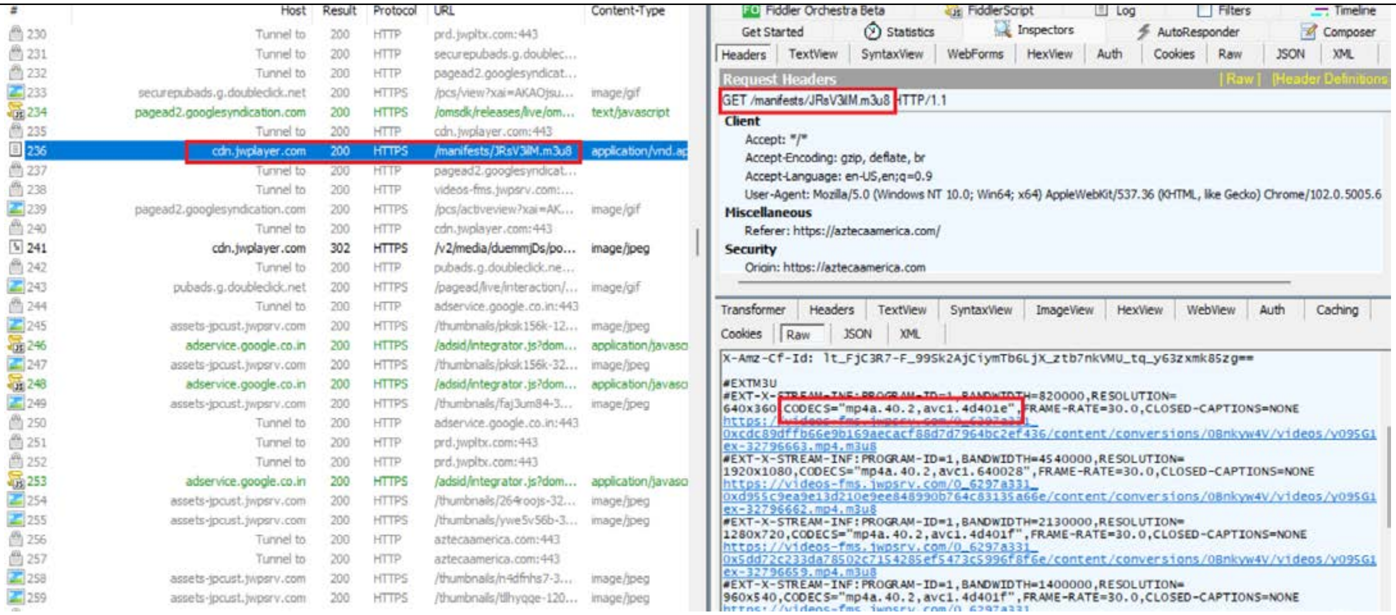
```
230   Tunnel to                        200  HTTP   prd.jwpltx.com:443
231   Tunnel to                        200  HTTP   securepubads.g.doublec...
232   Tunnel to                        200  HTTP   pagead2.googlesyndicat...
233   securepubads.g.doubleclick.net   200  HTTPS  /pcs/view?xai=AKAOjsu...       image/gif
234   pagead2.googlesyndication.com    200  HTTPS  /omsdk/releases/live/om...     text/javascript
235   Tunnel to                        200  HTTP   cdn.jwplayer.com:443
236   cdn.jwplayer.com                 200  HTTPS  /manifests/JRsV3lIM.m3u8       application/vnd.ap
237   Tunnel to                        200  HTTP   pagead2.googlesyndicat...
238   Tunnel to                        200  HTTP   videos-fms.jwpsrv.com:...
239   pagead2.googlesyndication.com    200  HTTPS  /pcs/activeview?xai=AK...      image/gif
240   Tunnel to                        200  HTTP   cdn.jwplayer.com:443
241   cdn.jwplayer.com                 302  HTTPS  /v2/media/duemmjDs/po...       image/jpeg
242   Tunnel to                        200  HTTP   pubads.g.doubleclick.ne...
243   pubads.g.doubleclick.net         200  HTTPS  /pagead/live/interaction/...   image/gif
244   Tunnel to                        200  HTTP   adservice.google.co.in:443
245   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/pksk156k-12...     image/jpeg
246   adservice.google.co.in           200  HTTPS  /adsid/integrator.js?dom...    application/javasc
247   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/pksk156k-32...     image/jpeg
248   adservice.google.co.in           200  HTTPS  /adsid/integrator.js?dom...    application/javasc
249   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/faj3um84-3...      image/jpeg
250   Tunnel to                        200  HTTP   adservice.google.co.in:443
251   Tunnel to                        200  HTTP   prd.jwpltx.com:443
252   Tunnel to                        200  HTTP   prd.jwpltx.com:443
253   adservice.google.co.in           200  HTTPS  /adsid/integrator.js?dom...    application/javasc
254   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/264roojs-32...     image/jpeg
255   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/ywe5v56b-3...      image/jpeg
256   Tunnel to                        200  HTTP   aztecaamerica.com:443
257   Tunnel to                        200  HTTP   aztecaamerica.com:443
258   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/n4dfnhs7-3...      image/jpeg
259   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/tlhyqqe-120...     image/jpeg

[QuickExec] ALT+Q > type HELP to learn more
```

```
Request Headers                                   [ Raw ]  [Header Definitions]
GET /manifests/JRsV3lIM.m3u8 HTTP/1.1
Client
    Accept: */*
    Accept-Encoding: gzip, deflate, br
    Accept-Language: en-US,en;q=0.9
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.5005.6
Miscellaneous
    Referer: https://aztecaamerica.com/
Security
    Origin: https://aztecaamerica.com
```

```
X-Amz-Cf-Id: 1t_FjC3R7-F_99Sk2AjCiymTb6LjX_ztb7nkVMU_tq_y63zxmk8Szg==

#EXTM3U
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=820000,RESOLUTION=
640x360,CODECS="mp4a.40.2,avc1.4d401e",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
0xcdc89dffb66e9b169aecacf88d7d7964bc2ef436/content/conversions/0Bnkyw4V/videos/y095G1
ex-32796663.mp4.m3u8
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=4540000,RESOLUTION=
1920x1080,CODECS="mp4a.40.2,avc1.640028",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
0xd955c9ea9e13d210e9ee848990b764c83135a66e/content/conversions/0Bnkyw4V/videos/y095G1
ex-32796662.mp4.m3u8
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=2130000,RESOLUTION=
1280x720,CODECS="mp4a.40.2,avc1.4d401f",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
0x5dd72c233da78502c7154285ef5473c5996f8f6e/content/conversions/0Bnkyw4V/videos/y095G1
ex-32796659.mp4.m3u8
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=1400000,RESOLUTION=
960x540,CODECS="mp4a.40.2,avc1.4d401f",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
```

Source: Packet capturing by fiddler tool

An Azteca America server streams video using mp4 based codec (e.g., video encoding) as shown below.



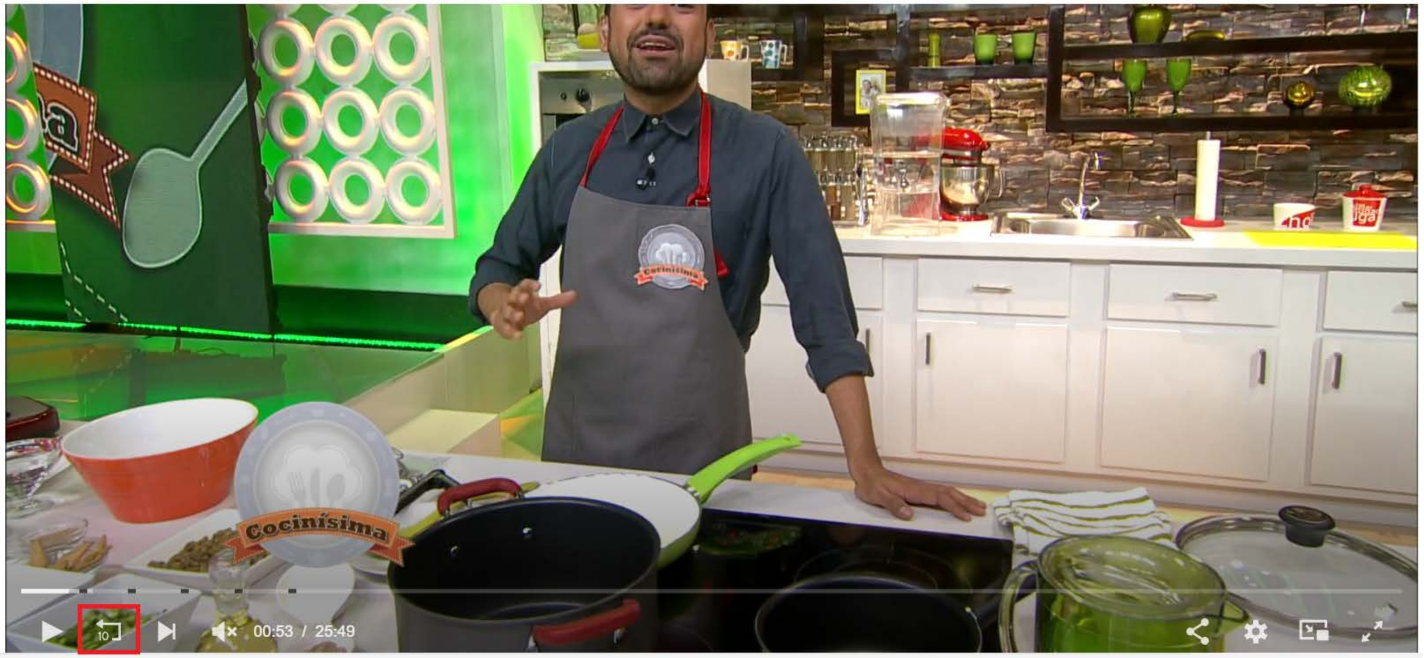
#	Host	Result	Protocol	URL	Content-Type
230	Tunnel to	200	HTTP	prd.jwpltx.com:443	
231	Tunnel to	200	HTTP	securepubads.g.doublec...	
232	Tunnel to	200	HTTP	pagead2.googlesyndicat...	
233	securepubads.g.doubleclick.net	200	HTTPS	/pcs/view?xai=AKAOjsu...	image/gif
234	pagead2.googlesyndication.com	200	HTTPS	/omsdk/releases/live/om...	text/javascript
235	Tunnel to	200	HTTP	cdn.jwplayer.com:443	
236	cdn.jwplayer.com	200	HTTPS	/manifests/JRsV3lM.m3u8	application/vnd.ap
237	Tunnel to	200	HTTP	pagead2.googlesyndicat...	
238	Tunnel to	200	HTTP	videos-fms.jwpsrv.com:...	
239	pagead2.googlesyndication.com	200	HTTPS	/pcs/activeview?xai=AK...	image/gif
240	Tunnel to	200	HTTP	cdn.jwplayer.com:443	
241	cdn.jwplayer.com	302	HTTPS	/v2/media/duemmjDs/po...	image/jpeg
242	Tunnel to	200	HTTP	pubads.g.doubleclick.ne...	
243	pubads.g.doubleclick.net	200	HTTPS	/pagead/live/interaction/...	image/gif
244	Tunnel to	200	HTTP	adservice.google.co.in:443	
245	assets-jpcust.jwpsrv.com	200	HTTPS	/thumbnails/pksk156k-12...	image/jpeg
246	adservice.google.co.in	200	HTTPS	/adsid/integrator.js?dom...	application/javasc
247	assets-jpcust.jwpsrv.com	200	HTTPS	/thumbnails/pksk156k-32...	image/jpeg
248	adservice.google.co.in	200	HTTPS	/adsid/integrator.js?dom...	application/javasc
249	assets-jpcust.jwpsrv.com	200	HTTPS	/thumbnails/faj3um84-3...	image/jpeg
250	Tunnel to	200	HTTP	adservice.google.co.in:443	
251	Tunnel to	200	HTTP	prd.jwpltx.com:443	
252	Tunnel to	200	HTTP	prd.jwpltx.com:443	
253	adservice.google.co.in	200	HTTPS	/adsid/integrator.js?dom...	application/javasc
254	assets-jpcust.jwpsrv.com	200	HTTPS	/thumbnails/264roojs-32...	image/jpeg
255	assets-jpcust.jwpsrv.com	200	HTTPS	/thumbnails/ywe5v56b-3...	image/jpeg
256	Tunnel to	200	HTTP	aztecaamerica.com:443	
257	Tunnel to	200	HTTP	aztecaamerica.com:443	
258	assets-jpcust.jwpsrv.com	200	HTTPS	/thumbnails/n4dfnhs7-3...	image/jpeg
259	assets-jpcust.jwpsrv.com	200	HTTPS	/thumbnails/tlhyqqe-120...	image/jpeg

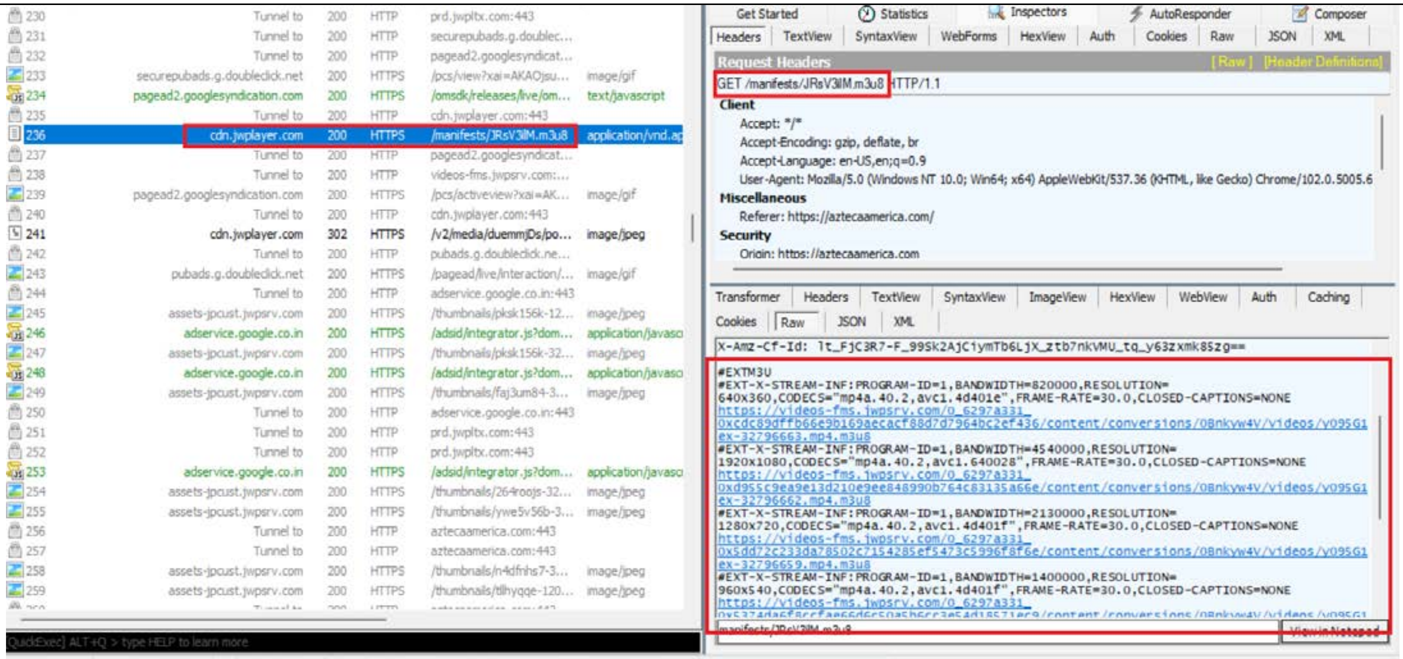
```
Request Headers                                   [ Raw ]  [Header Definitions]
GET /manifests/JRsV3lM.m3u8 HTTP/1.1
Client
    Accept: */*
    Accept-Encoding: gzip, deflate, br
    Accept-Language: en-US,en;q=0.9
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.5005.6
Miscellaneous
    Referer: https://aztecaamerica.com/
Security
    Origin: https://aztecaamerica.com
```

```
X-Amz-Cf-Id: lt_FjC3R7-F_99Sk2AjCiymTb6LjX_ztb7nkVMU_tq_y63zxmk85zg==

#EXTM3U
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=820000,RESOLUTION=
640x360,CODECS="mp4a.40.2,avc1.4d401e",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
0xcdc89dffb66e9b169aecacf88d7d7964bc2ef436/content/conversions/0Bnkyw4V/videos/y095G1
ex-32796663.mp4.m3u8
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=4540000,RESOLUTION=
1920x1080,CODECS="mp4a.40.2,avc1.640028",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
0xd955c9ea9e13d210e9ee848990b764c83135a66e/content/conversions/0Bnkyw4V/videos/y095G1
ex-32796662.mp4.m3u8
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=2130000,RESOLUTION=
1280x720,CODECS="mp4a.40.2,avc1.4d401f",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
0x5dd72c233da78502c7154285ef5473c5996f8f6e/content/conversions/0Bnkyw4V/videos/y095G1
ex-32796659.mp4.m3u8
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=1400000,RESOLUTION=
960x540,CODECS="mp4a.40.2,avc1.4d401f",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331
```

Source: Packet capturing by fiddler tool

Azteca America streams videos with the capability of being played with trick mode (e.g., 10 second reverse playback, etc.)

	 https://aztecaamerica.com/ An Azteca America server sends references (URLs) of all the chunks of the video to be streamed.

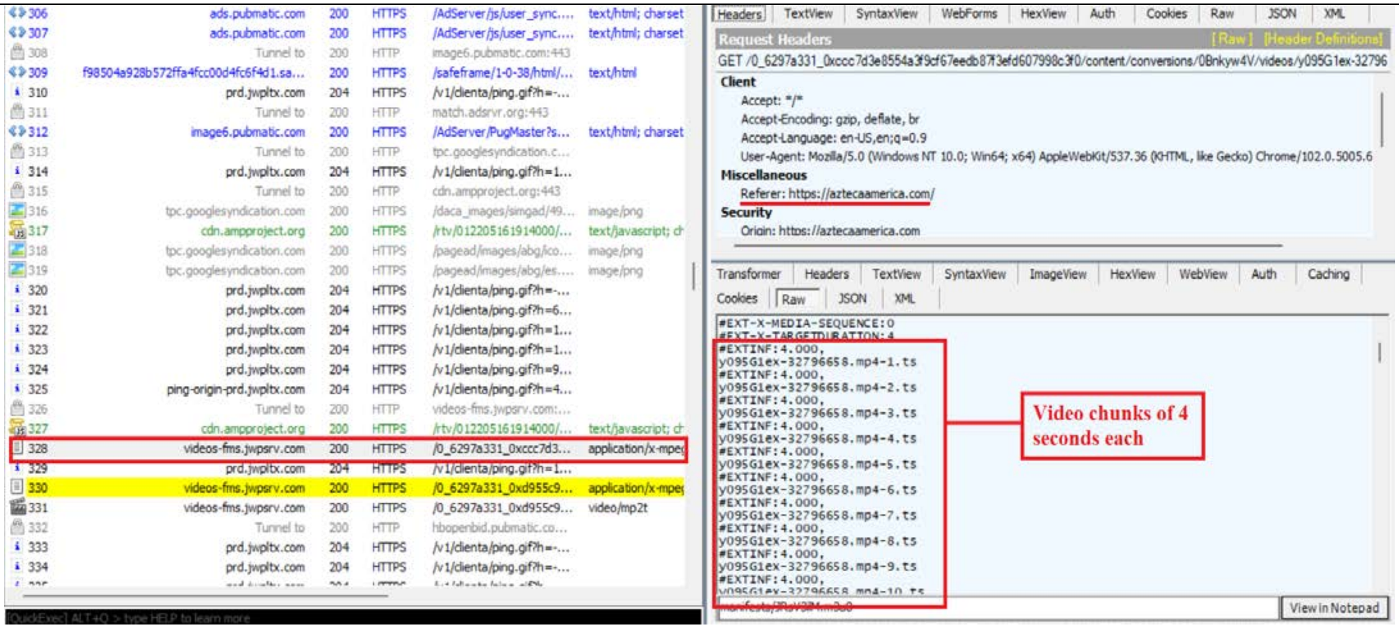


```
230   Tunnel to                        200  HTTP   prd.jwpltx.com:443
231   Tunnel to                        200  HTTP   securepubads.g.doublec...
232   Tunnel to                        200  HTTP   pagead2.googlesyndicat...
233   securepubads.g.doubleclick.net   200  HTTPS  /pcs/view?xai=AKAOjsu...       image/gif
234   pagead2.googlesyndication.com    200  HTTPS  /omsdk/releases/live/om...     text/javascript
235   Tunnel to                        200  HTTP   cdn.jwplayer.com:443
236   cdn.jwplayer.com                 200  HTTPS  /manifests/JRsV3lIM.m3u8       application/vnd.ap
237   Tunnel to                        200  HTTP   pagead2.googlesyndicat...
238   Tunnel to                        200  HTTP   videos-fms.jwpsrv.com:...
239   pagead2.googlesyndication.com    200  HTTPS  /pcs/activeview?xai=AK...      image/gif
240   Tunnel to                        200  HTTP   cdn.jwplayer.com:443
241   cdn.jwplayer.com                 302  HTTPS  /v2/media/duemmjDs/po...       image/jpeg
242   Tunnel to                        200  HTTP   pubads.g.doubleclick.ne...
243   pubads.g.doubleclick.net         200  HTTPS  /pagead/live/interaction/...   image/gif
244   Tunnel to                        200  HTTP   adservice.google.co.in:443
245   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/pksk156k-12...     image/jpeg
246   adservice.google.co.in           200  HTTPS  /adsid/integrator.js?dom...    application/javasc
247   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/pksk156k-32...     image/jpeg
248   adservice.google.co.in           200  HTTPS  /adsid/integrator.js?dom...    application/javasc
249   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/faj3um84-3...      image/jpeg
250   Tunnel to                        200  HTTP   adservice.google.co.in:443
251   Tunnel to                        200  HTTP   prd.jwpltx.com:443
252   Tunnel to                        200  HTTP   prd.jwpltx.com:443
253   adservice.google.co.in           200  HTTPS  /adsid/integrator.js?dom...    application/javasc
254   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/264roojs-32...     image/jpeg
255   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/ywe5v56b-3...      image/jpeg
256   Tunnel to                        200  HTTP   aztecaamerica.com:443
257   Tunnel to                        200  HTTP   aztecaamerica.com:443
258   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/n4dfnhs7-3...      image/jpeg
259   assets-jpcust.jwpsrv.com         200  HTTPS  /thumbnails/tllhyqqe-120...    image/jpeg

[QuickExec] ALT+Q > type HELP to learn more
```

```
GET /manifests/JRsV3lIM.m3u8 HTTP/1.1
Client
    Accept: */*
    Accept-Encoding: gzip, deflate, br
    Accept-Language: en-US,en;q=0.9
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.5005.6
Miscellaneous
    Referer: https://aztecaamerica.com/
Security
    Origin: https://aztecaamerica.com
```

```
X-Amz-Cf-Id: 1t_FjC3R7-F_99Sk2AjCiymTb6LjX_ztb7nkVMU_tq_y63zxmk8Szg==

#EXTM3U
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=820000,RESOLUTION=
640x360,CODECS="mp4a.40.2,avc1.4d401e",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
0xcdc89dffb66e9b169aecacf88d7d7964bc2ef436/content/conversions/08nkyw4V/videos/y095G1
ex-32796663.mp4.m3u8
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=4540000,RESOLUTION=
1920x1080,CODECS="mp4a.40.2,avc1.640028",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
0xd955c9ea9e13d210e9ee848990b764c83135a66e/content/conversions/08nkyw4V/videos/y095G1
ex-32796662.mp4.m3u8
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=2130000,RESOLUTION=
1280x720,CODECS="mp4a.40.2,avc1.4d401f",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
0x5dd72c233da78502c7154285ef5473c5996f8f6e/content/conversions/08nkyw4V/videos/y095G1
ex-32796659.mp4.m3u8
#EXT-X-STREAM-INF:PROGRAM-ID=1,BANDWIDTH=1400000,RESOLUTION=
960x540,CODECS="mp4a.40.2,avc1.4d401f",FRAME-RATE=30.0,CLOSED-CAPTIONS=NONE
https://videos-fms.jwpsrv.com/0_6297a331_
[illegible]/content/conversions/08nkyw4V/videos/y095G1
manifests/JRsV3lIM.m3u8
```

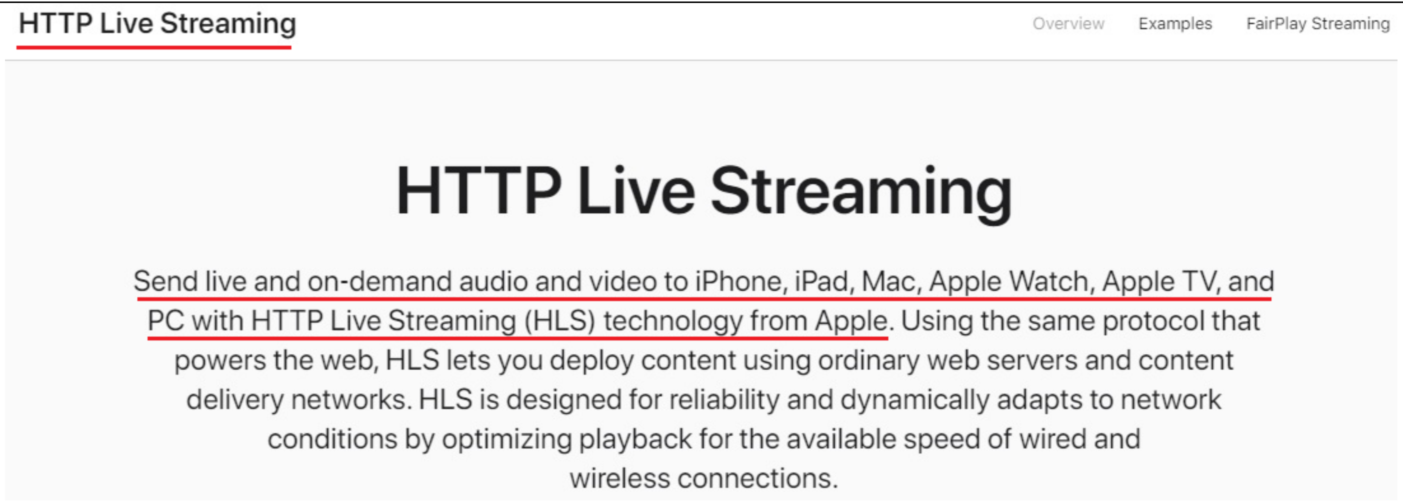
Source: Packet capturing by fiddler tool

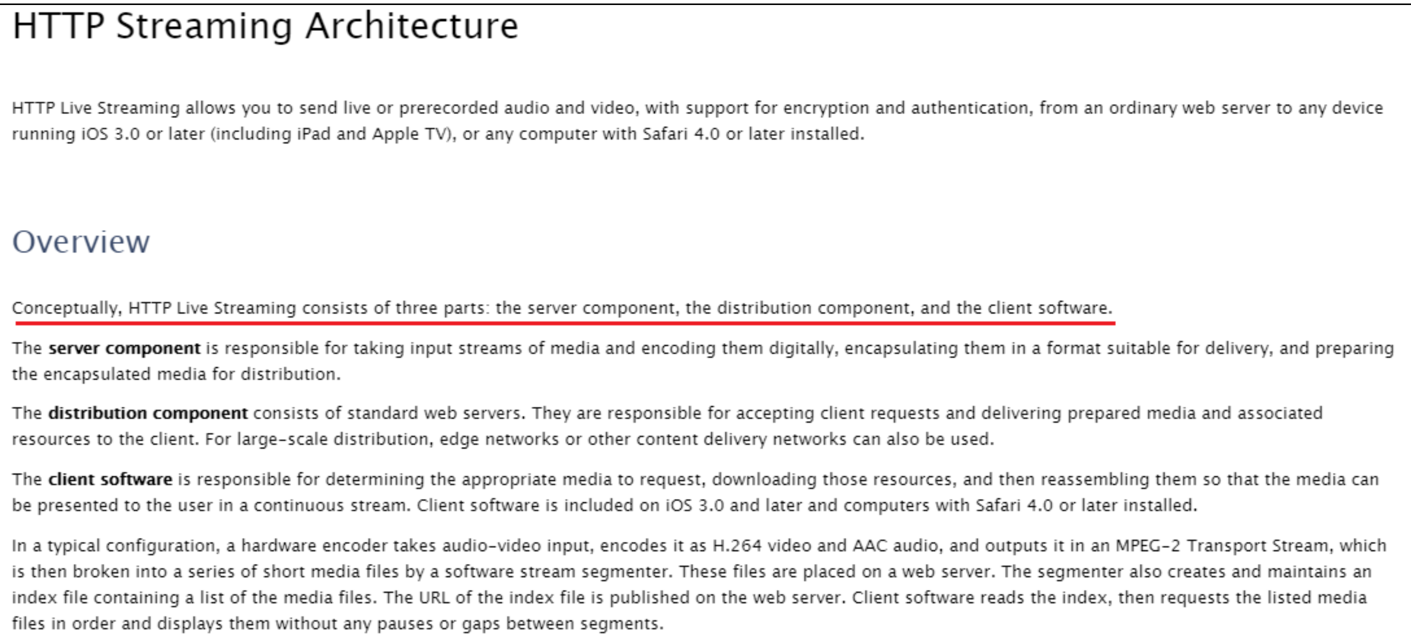
	 Source: Packet capturing by fiddler tool

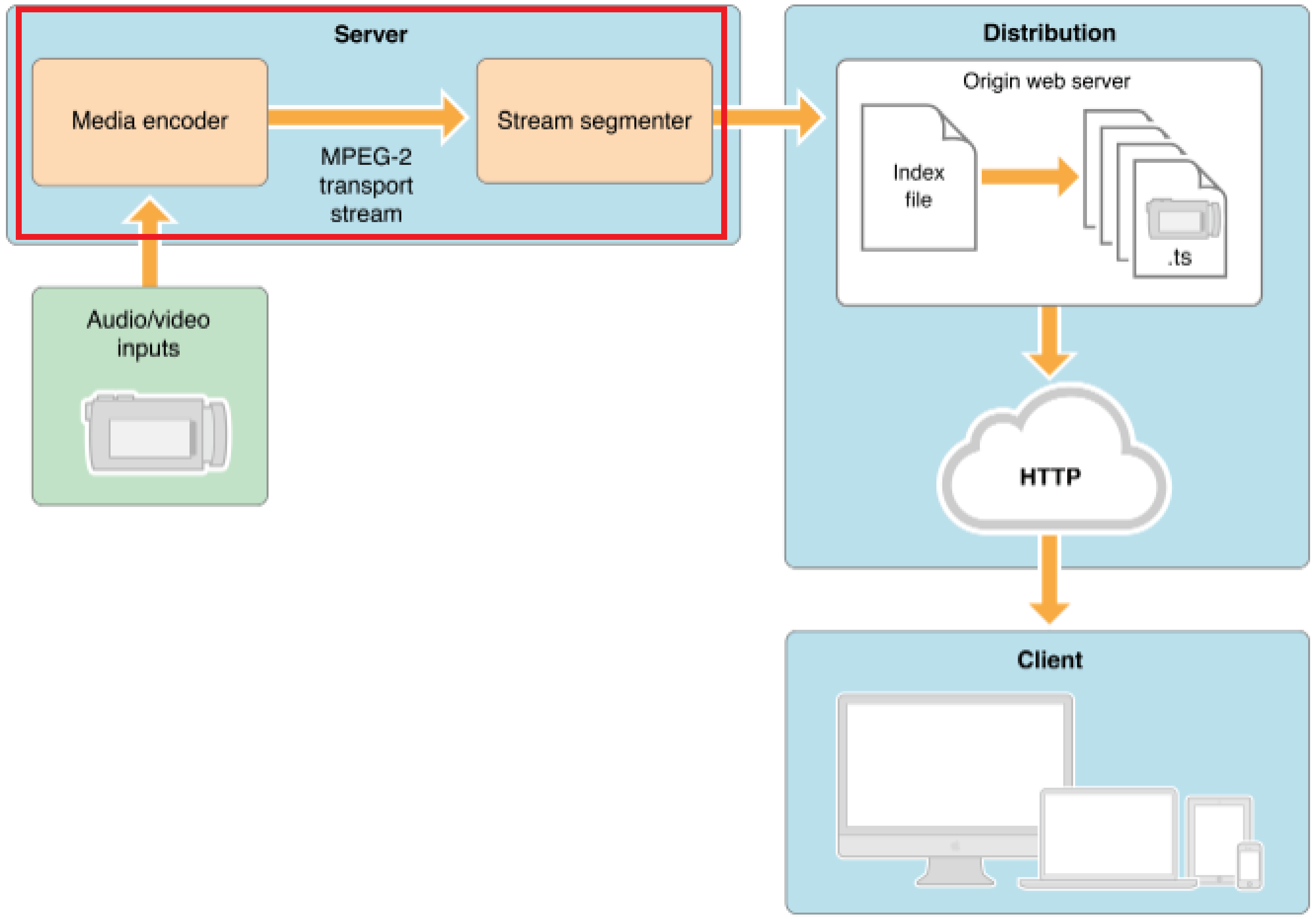
```
306  ads.pubmatic.com                          200  HTTPS  /AdServer/js/user_sync....   text/html; charset
307  ads.pubmatic.com                          200  HTTPS  /AdServer/js/user_sync....   text/html; charset
308  Tunnel to                                 200  HTTP   image6.pubmatic.com:443
309  f98504a928b572ffa4fcc00d4fc6f4d1.sa...    200  HTTPS  /safeframe/1-0-38/html/...   text/html
310  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=-...
311  Tunnel to                                 200  HTTP   match.adsrvr.org:443
312  image6.pubmatic.com                       200  HTTPS  /AdServer/PugMaster?s...     text/html; charset
313  Tunnel to                                 200  HTTP   tpc.googlesyndication.c...
314  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=1...
315  Tunnel to                                 200  HTTP   cdn.ampproject.org:443
316  tpc.googlesyndication.com                 200  HTTPS  /daca_images/simgad/49...    image/png
317  cdn.ampproject.org                        200  HTTPS  /rtv/012205161914000/...     text/javascript; ch
318  tpc.googlesyndication.com                 200  HTTPS  /pagead/images/abg/ico...    image/png
319  tpc.googlesyndication.com                 200  HTTPS  /pagead/images/abg/es....    image/png
320  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=-...
321  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=6...
322  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=1...
323  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=1...
324  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=9...
325  ping-origin-prd.jwpltx.com                204  HTTPS  /v1/clienta/ping.gif?h=4...
326  Tunnel to                                 200  HTTP   videos-fms.jwpsrv.com:...
327  cdn.ampproject.org                        200  HTTPS  /rtv/012205161914000/...     text/javascript; ch
328  videos-fms.jwpsrv.com                     200  HTTPS  /0_6297a331_0xccc7d3...      application/x-mpeg
329  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=1...
330  videos-fms.jwpsrv.com                     200  HTTPS  /0_6297a331_0xd955c9...      application/x-mpeg
331  videos-fms.jwpsrv.com                     200  HTTPS  /0_6297a331_0xd955c9...      video/mp2t
332  Tunnel to                                 200  HTTP   hbopenbid.pubmatic.co...
333  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=-...
334  prd.jwpltx.com                            204  HTTPS  /v1/clienta/ping.gif?h=-...
[QuickExec] ALT+Q > type HELP to learn more
```

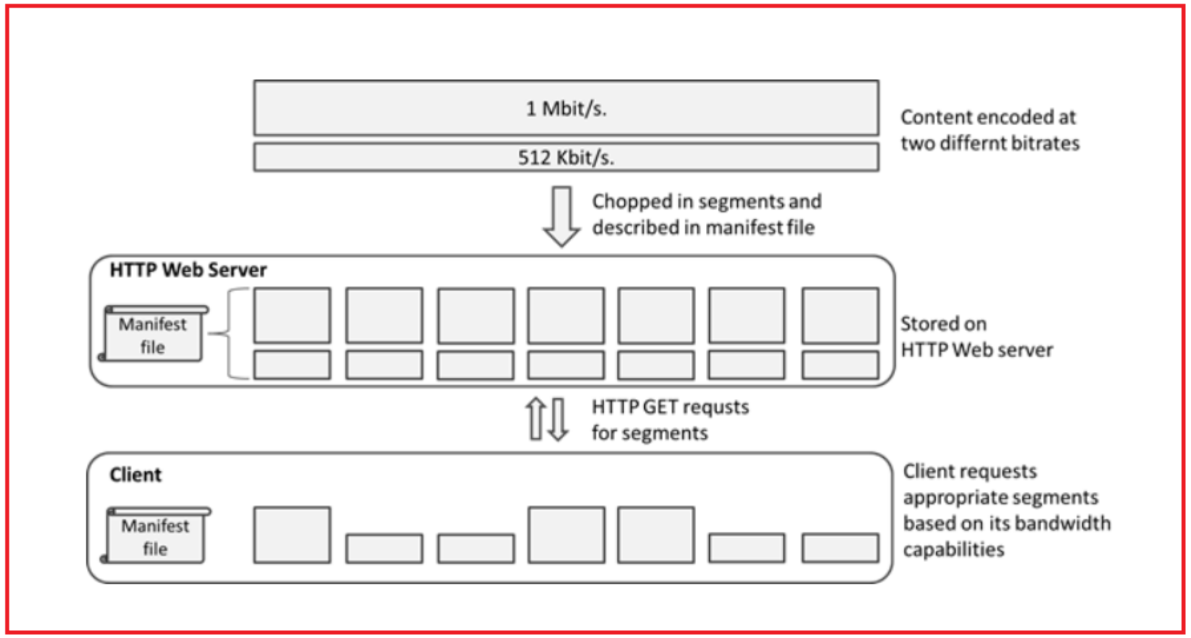
```
Request Headers
GET /0_6297a331_0xccc7d3e8554a3f9cf67eedb87f3efd607998c3f0/content/conversions/0Bnkyw4V/videos/y095G1ex-32796
Client
    Accept: */*
    Accept-Encoding: gzip, deflate, br
    Accept-Language: en-US,en;q=0.9
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.5005.6
Miscellaneous
    Referer: https://aztecaamerica.com/
Security
    Origin: https://aztecaamerica.com
```

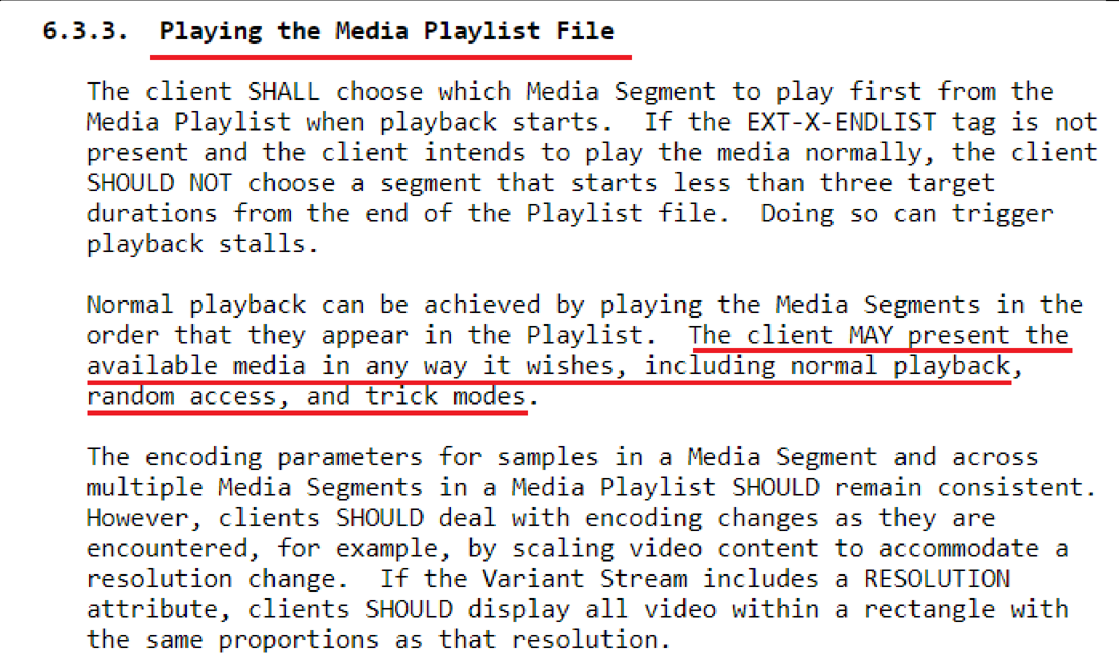
```
#EXT-X-MEDIA-SEQUENCE:0
#EXT-X-TARGETDURATION:4
#EXTINF:4.000,
y095G1ex-32796658.mp4-1.ts
#EXTINF:4.000,
y095G1ex-32796658.mp4-2.ts
#EXTINF:4.000,
y095G1ex-32796658.mp4-3.ts
#EXTINF:4.000,
y095G1ex-32796658.mp4-4.ts
#EXTINF:4.000,
y095G1ex-32796658.mp4-5.ts
#EXTINF:4.000,
y095G1ex-32796658.mp4-6.ts
#EXTINF:4.000,
y095G1ex-32796658.mp4-7.ts
#EXTINF:4.000,
y095G1ex-32796658.mp4-8.ts
#EXTINF:4.000,
y095G1ex-32796658.mp4-9.ts
#EXTINF:4.000,
y095G1ex-32796658.mp4-10.ts
```

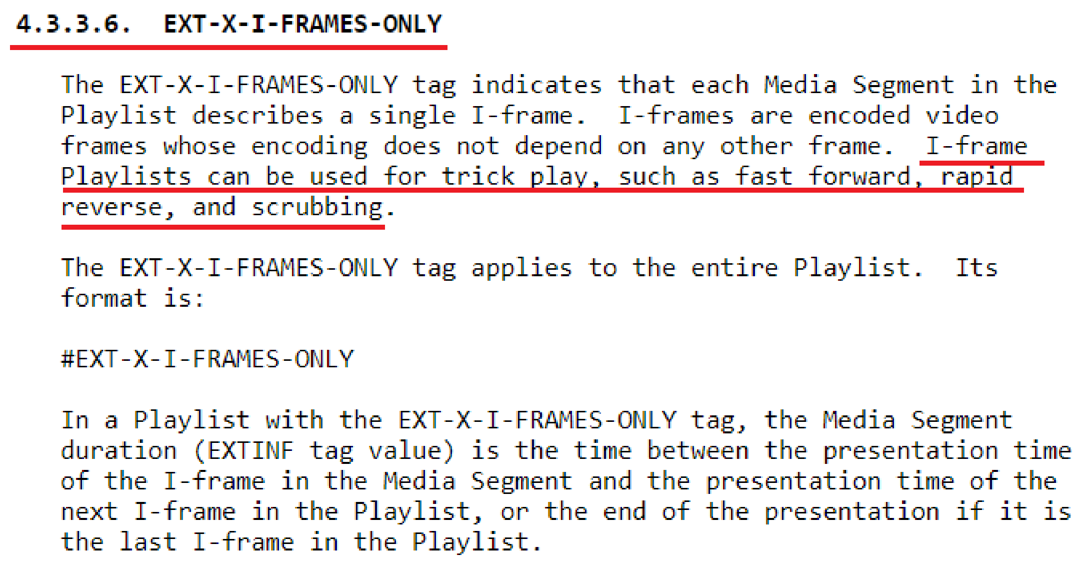
	 HTTP Live Streaming Overview Examples FairPlay Streaming HTTP Live Streaming Send live and on-demand audio and video to iPhone, iPad, Mac, Apple Watch, Apple TV, and PC with HTTP Live Streaming (HLS) technology from Apple. Using the same protocol that powers the web, HLS lets you deploy content using ordinary web servers and content delivery networks. HLS is designed for reliability and dynamically adapts to network conditions by optimizing playback for the available speed of wired and wireless connections. https://developer.apple.com/streaming/

	 HTTP Streaming Architecture HTTP Live Streaming allows you to send live or prerecorded audio and video, with support for encryption and authentication, from an ordinary web server to any device running iOS 3.0 or later (including iPad and Apple TV), or any computer with Safari 4.0 or later installed. Overview Conceptually, HTTP Live Streaming consists of three parts: the server component, the distribution component, and the client software. The **server component** is responsible for taking input streams of media and encoding them digitally, encapsulating them in a format suitable for delivery, and preparing the encapsulated media for distribution. The **distribution component** consists of standard web servers. They are responsible for accepting client requests and delivering prepared media and associated resources to the client. For large-scale distribution, edge networks or other content delivery networks can also be used. The **client software** is responsible for determining the appropriate media to request, downloading those resources, and then reassembling them so that the media can be presented to the user in a continuous stream. Client software is included on iOS 3.0 and later and computers with Safari 4.0 or later installed. In a typical configuration, a hardware encoder takes audio-video input, encodes it as H.264 video and AAC audio, and outputs it in an MPEG-2 Transport Stream, which is then broken into a series of short media files by a software stream segmenter. These files are placed on a web server. The segmenter also creates and maintains an index file containing a list of the media files. The URL of the index file is published on the web server. Client software reads the index, then requests the listed media files in order and displays them without any pauses or gaps between segments. https://developer.apple.com/library/archive/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/HTTPStreamingArchitecture/HTTPStreamingArchitecture.html#//apple_ref/doc/uid/TP40008332-CH101-SW4

	 Server Media encoder MPEG-2 transport stream Stream segmenter Audio/video inputs Distribution Origin web server Index file .ts HTTP Client https://developer.apple.com/library/archive/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/HTTPStreamingArchitecture/HTTPStreamingArchitecture.html#//apple_ref/doc/uid/TP40008332-CH101-SW4

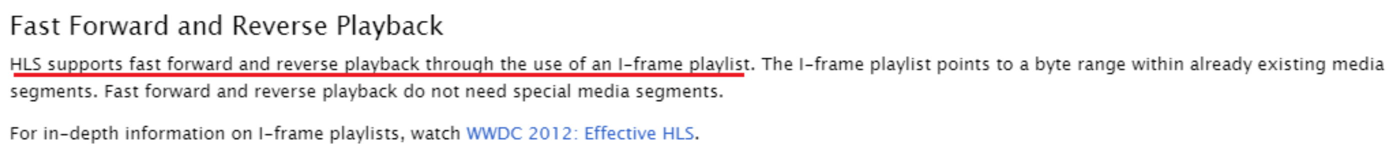
	 1 Mbit/s. 512 Kbit/s. Content encoded at two differnt bitrates Chopped in segments and described in manifest file HTTP Web Server Manifest file Stored on HTTP Web server HTTP GET requsts for segments Client Manifest file Client requests appropriate segments based on its bandwidth capabilities http://blog.jpauli.tech/live-video-streaming-stack-from-home/

	 6.3.3. **Playing the Media Playlist File** The client SHALL choose which Media Segment to play first from the Media Playlist when playback starts. If the EXT-X-ENDLIST tag is not present and the client intends to play the media normally, the client SHOULD NOT choose a segment that starts less than three target durations from the end of the Playlist file. Doing so can trigger playback stalls. Normal playback can be achieved by playing the Media Segments in the order that they appear in the Playlist. The client MAY present the available media in any way it wishes, including normal playback, random access, and trick modes. The encoding parameters for samples in a Media Segment and across multiple Media Segments in a Media Playlist SHOULD remain consistent. However, clients SHOULD deal with encoding changes as they are encountered, for example, by scaling video content to accommodate a resolution change. If the Variant Stream includes a RESOLUTION attribute, clients SHOULD display all video within a rectangle with the same proportions as that resolution. https://tools.ietf.org/html/rfc8216



```
4.3.3.6.  EXT-X-I-FRAMES-ONLY

   The EXT-X-I-FRAMES-ONLY tag indicates that each Media Segment in the
   Playlist describes a single I-frame.  I-frames are encoded video
   frames whose encoding does not depend on any other frame.  I-frame
   Playlists can be used for trick play, such as fast forward, rapid
   reverse, and scrubbing.

   The EXT-X-I-FRAMES-ONLY tag applies to the entire Playlist.  Its
   format is:

   #EXT-X-I-FRAMES-ONLY

   In a Playlist with the EXT-X-I-FRAMES-ONLY tag, the Media Segment
   duration (EXTINF tag value) is the time between the presentation time
   of the I-frame in the Media Segment and the presentation time of the
   next I-frame in the Playlist, or the end of the presentation if it is
   the last I-frame in the Playlist.
```

https://tools.ietf.org/html/rfc8216

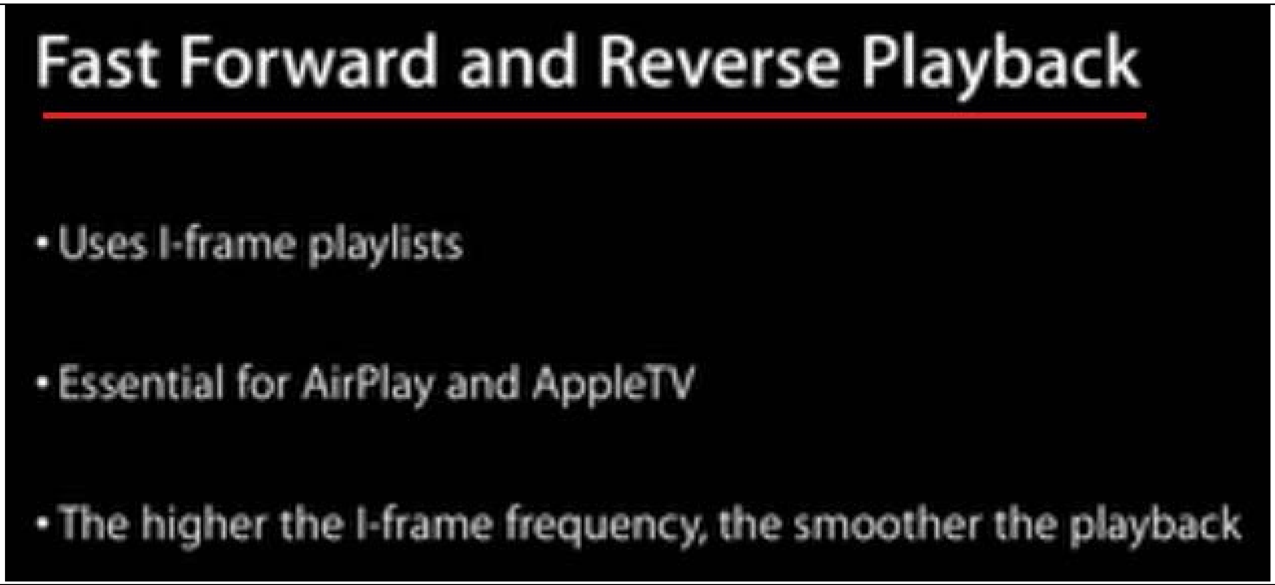


Fast Forward and Reverse Playback

HLS supports fast forward and reverse playback through the use of an I-frame playlist. The I-frame playlist points to a byte range within already existing media segments. Fast forward and reverse playback do not need special media segments.

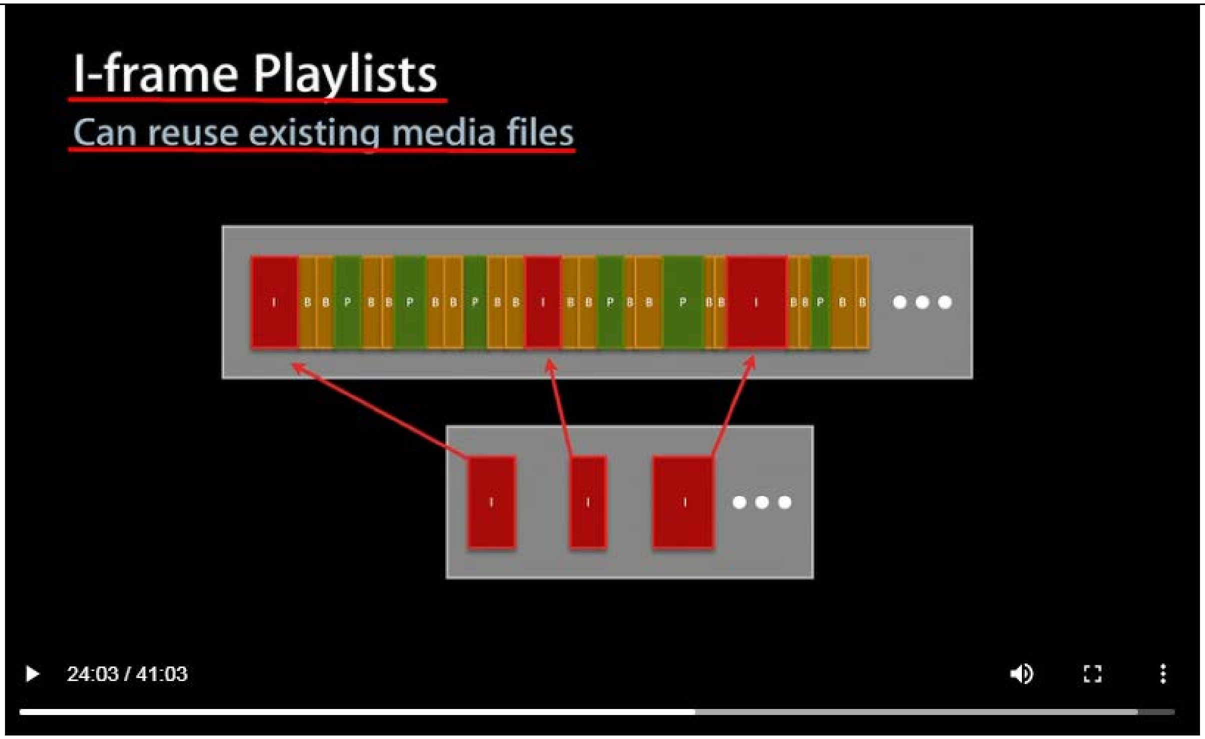
For in-depth information on I-frame playlists, watch WWDC 2012: Effective HLS.

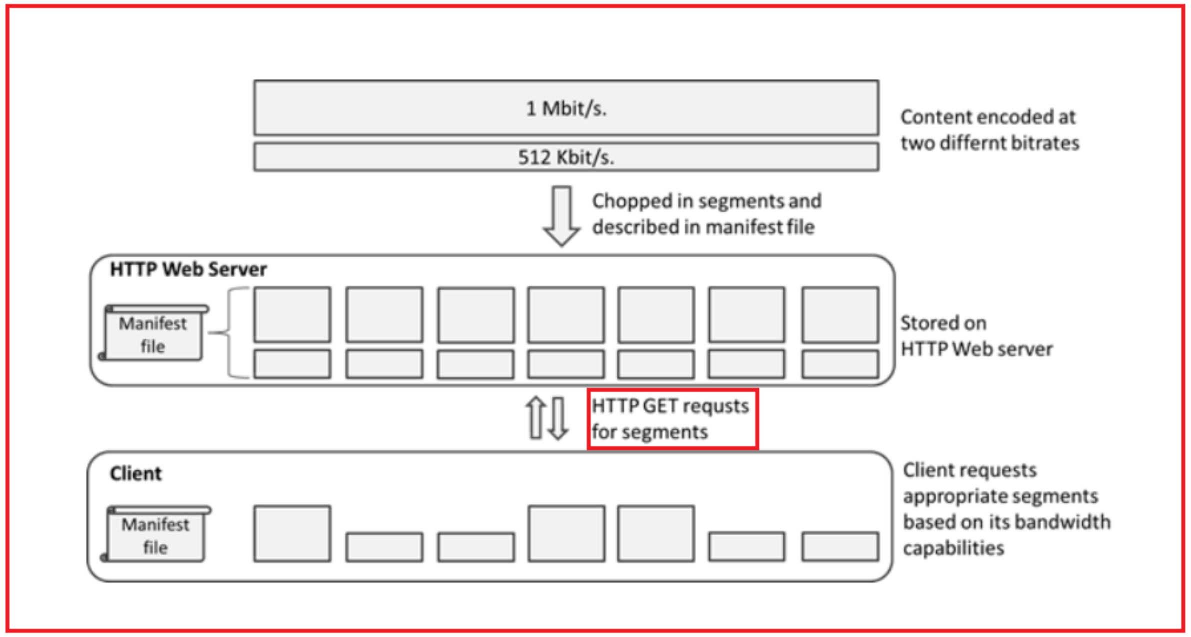
https://developer.apple.com/library/archive/referencelibrary/GettingStarted/AboutHTTPLiveStreaming/about/about.html

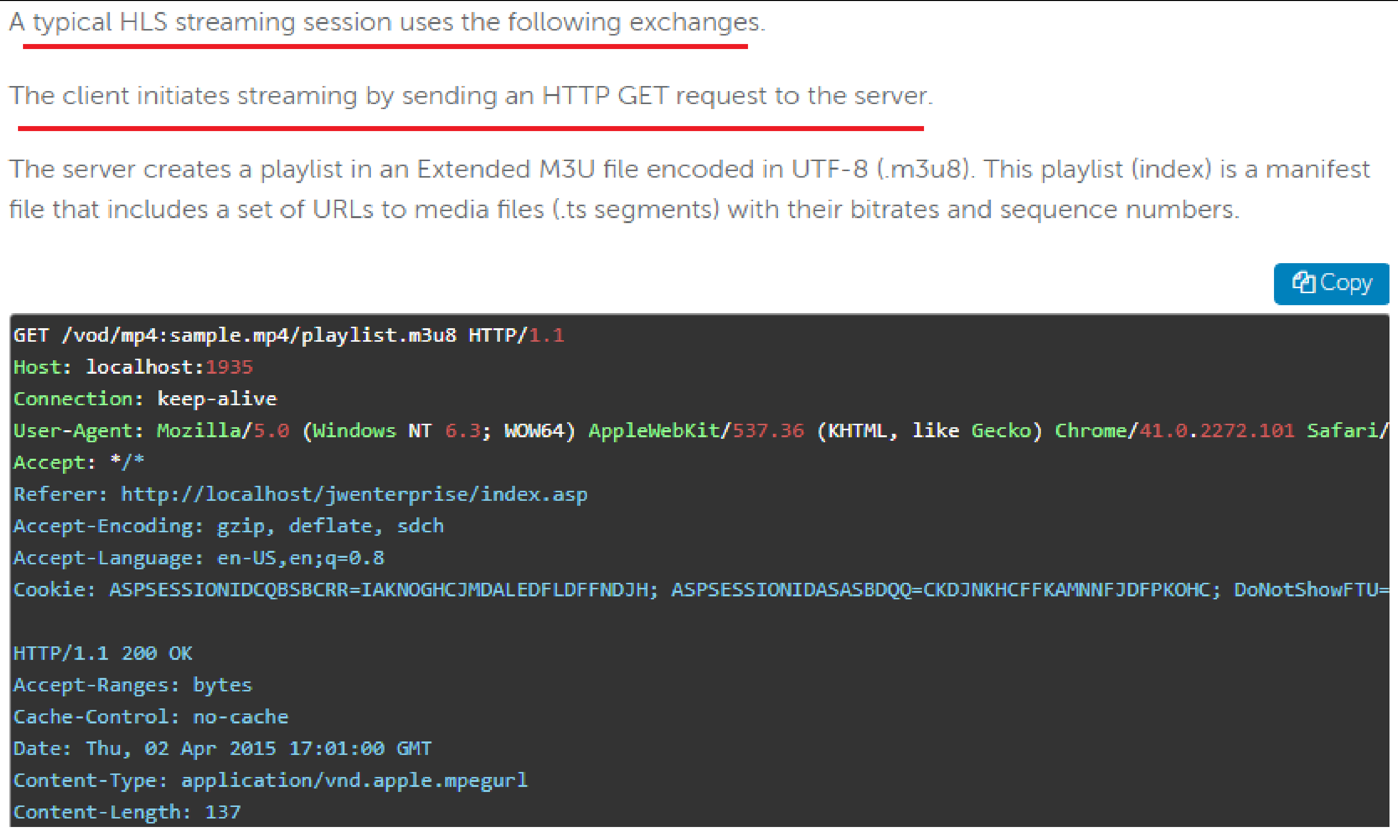
	

Fast Forward and Reverse Playback

- Uses I-frame playlists
- Essential for AirPlay and AppleTV
- The higher the I-frame frequency, the smoother the playback

	 I-frame Playlists Can reuse existing media files 24:03 / 41:03 https://developer.apple.com/videos/play/wwdc2012/502/
creation of an HTTP GET request for requesting a fast search operation of an original video stream, the request stating a playback speed parameter and an initial position and optionally at least one parameter selected from a	The Standard practices creation of an HTTP GET request (e.g., an HTTP request by the HLS client) for requesting a fast search operation (e.g., trick mode operation such as fast forward or rewind) of an original video stream (e.g., a video stream), the request stating a playback speed parameter (e.g., playout rate) and an initial position (e.g., the media sequence number of the next segment to load when the trick mode is requested) and optionally at least one parameter selected from a group of parameters consisting of file name, file type, path (e.g., a URL of requested segments for trick mode operation), and playback direction.

group of parameters consisting of file name, file type, path, and playback direction;	 1 Mbit/s. 512 Kbit/s. Content encoded at two differnt bitrates Chopped in segments and described in manifest file HTTP Web Server Manifest file Stored on HTTP Web server HTTP GET requsts for segments Client Manifest file Client requests appropriate segments based on its bandwidth capabilities http://blog.jpauli.tech/live-video-streaming-stack-from-home/ As shown below, a client sends an HTTP Get request to an HLS media segment distribution server for media segments with URL of the media segments. Exemplary requests are shown below.

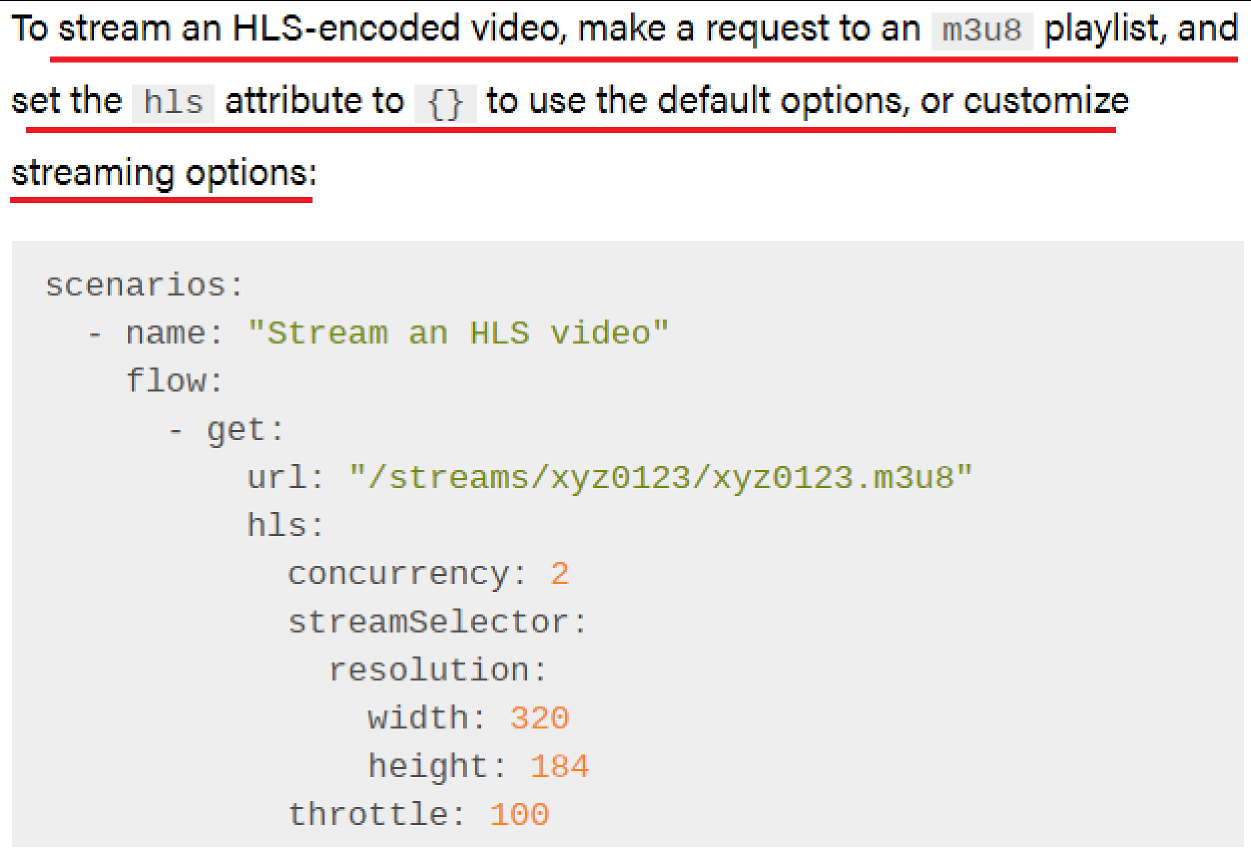

A typical HLS streaming session uses the following exchanges.

The client initiates streaming by sending an HTTP GET request to the server.

The server creates a playlist in an Extended M3U file encoded in UTF-8 (.m3u8). This playlist (index) is a manifest file that includes a set of URLs to media files (.ts segments) with their bitrates and sequence numbers.

```
GET /vod/mp4:sample.mp4/playlist.m3u8 HTTP/1.1
Host: localhost:1935
Connection: keep-alive
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2272.101 Safari/
Accept: */*
Referer: http://localhost/jwenterprise/index.asp
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8
Cookie: ASPSESSIONIDCQBSBCRR=IAKNOGHCJMDALEDFLDFFNDJH; ASPSESSIONIDASASBDQQ=CKDJNKHCFFKAMNNFJDFPKOHC; DoNotShowFTU=

HTTP/1.1 200 OK
Accept-Ranges: bytes
Cache-Control: no-cache
Date: Thu, 02 Apr 2015 17:01:00 GMT
Content-Type: application/vnd.apple.mpegurl
Content-Length: 137
```

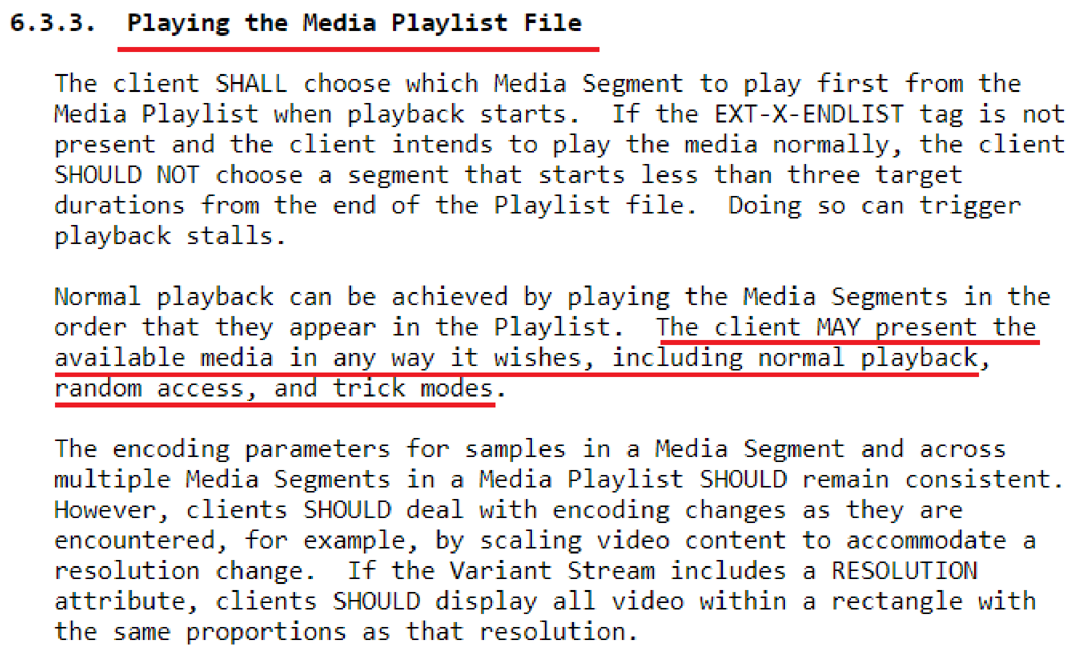
https://www.wowza.com/docs/how-to-troubleshoot-apple-hls-playback

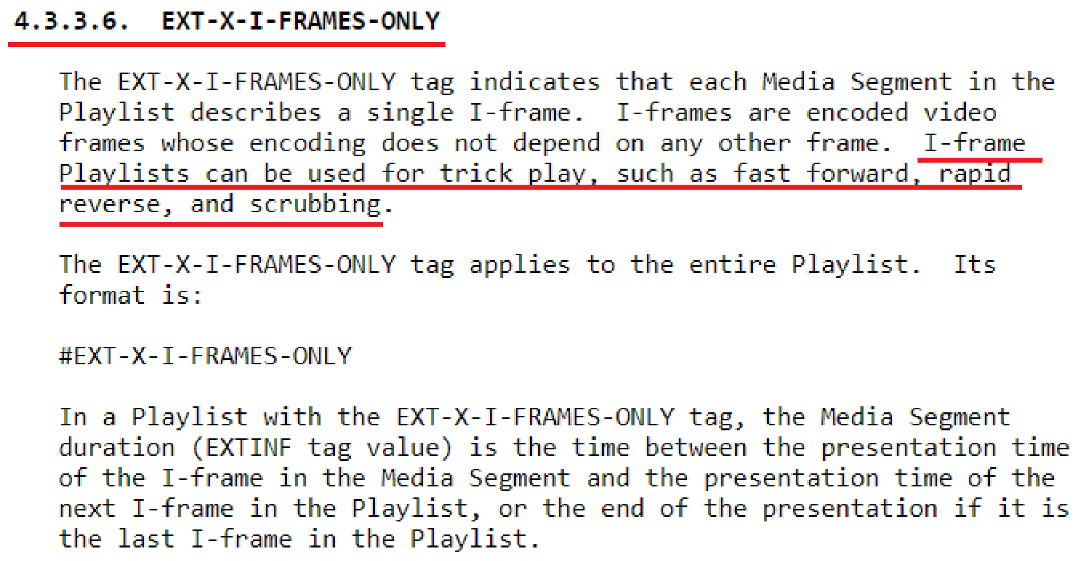


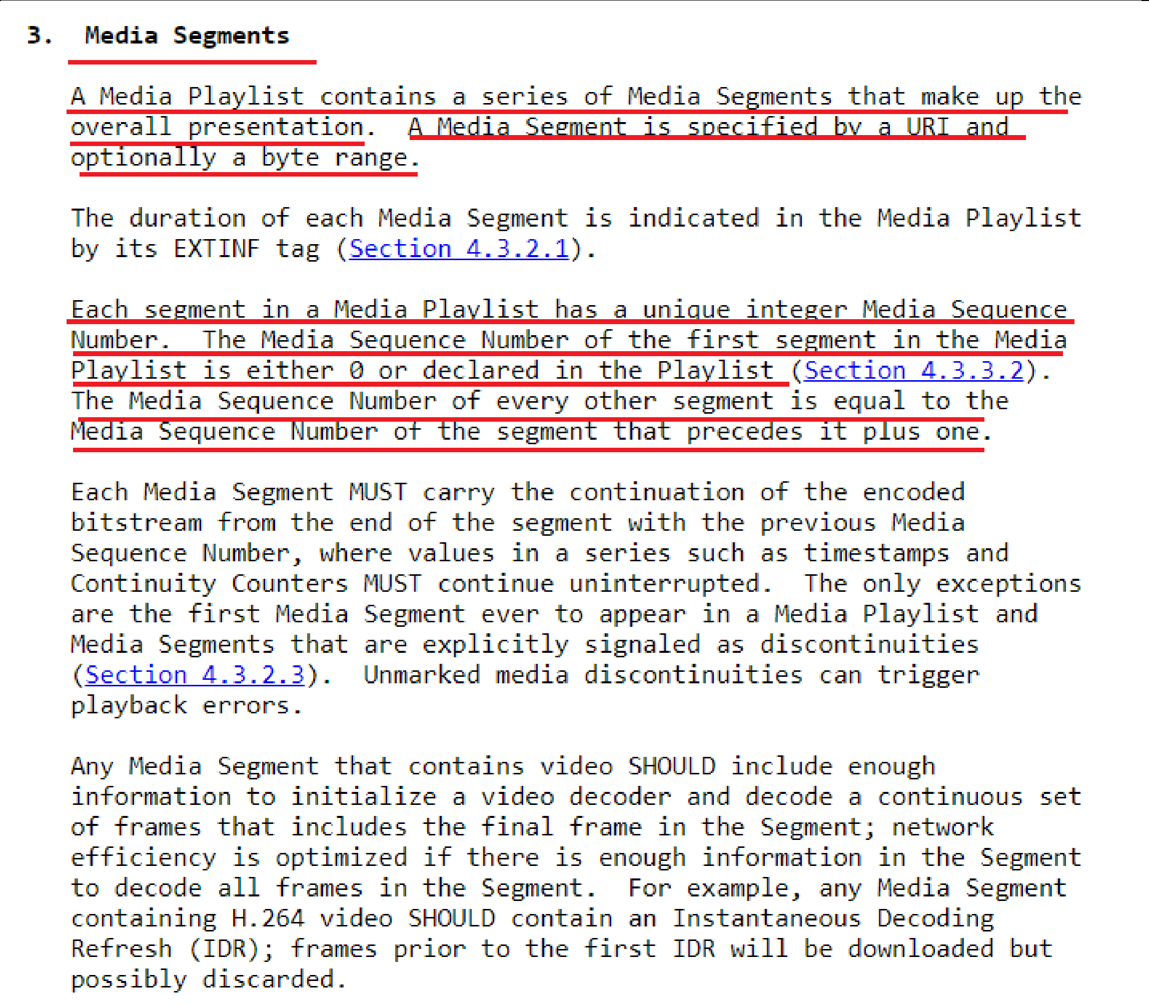
To stream an HLS-encoded video, make a request to an `m3u8` playlist, and set the `hls` attribute to `{}` to use the default options, or customize streaming options:

```
scenarios:
  - name: "Stream an HLS video"
    flow:
      - get:
          url: "/streams/xyz0123/xyz0123.m3u8"
          hls:
            concurrency: 2
            streamSelector:
              resolution:
                width: 320
                height: 184
            throttle: 100
```

https://artillery.io/docs/guides/plugins/plugin-hls.html

	 **6.3.3. Playing the Media Playlist File** The client SHALL choose which Media Segment to play first from the Media Playlist when playback starts. If the EXT-X-ENDLIST tag is not present and the client intends to play the media normally, the client SHOULD NOT choose a segment that starts less than three target durations from the end of the Playlist file. Doing so can trigger playback stalls. Normal playback can be achieved by playing the Media Segments in the order that they appear in the Playlist. The client MAY present the available media in any way it wishes, including normal playback, random access, and trick modes. The encoding parameters for samples in a Media Segment and across multiple Media Segments in a Media Playlist SHOULD remain consistent. However, clients SHOULD deal with encoding changes as they are encountered, for example, by scaling video content to accommodate a resolution change. If the Variant Stream includes a RESOLUTION attribute, clients SHOULD display all video within a rectangle with the same proportions as that resolution. https://tools.ietf.org/html/rfc8216

	 **4.3.3.6. EXT-X-I-FRAMES-ONLY** The EXT-X-I-FRAMES-ONLY tag indicates that each Media Segment in the Playlist describes a single I-frame. I-frames are encoded video frames whose encoding does not depend on any other frame. I-frame Playlists can be used for trick play, such as fast forward, rapid reverse, and scrubbing. The EXT-X-I-FRAMES-ONLY tag applies to the entire Playlist. Its format is: #EXT-X-I-FRAMES-ONLY In a Playlist with the EXT-X-I-FRAMES-ONLY tag, the Media Segment duration (EXTINF tag value) is the time between the presentation time of the I-frame in the Media Segment and the presentation time of the next I-frame in the Playlist, or the end of the presentation if it is the last I-frame in the Playlist. https://tools.ietf.org/html/rfc8216



3. Media Segments

A Media Playlist contains a series of Media Segments that make up the overall presentation. A Media Segment is specified by a URI and optionally a byte range.

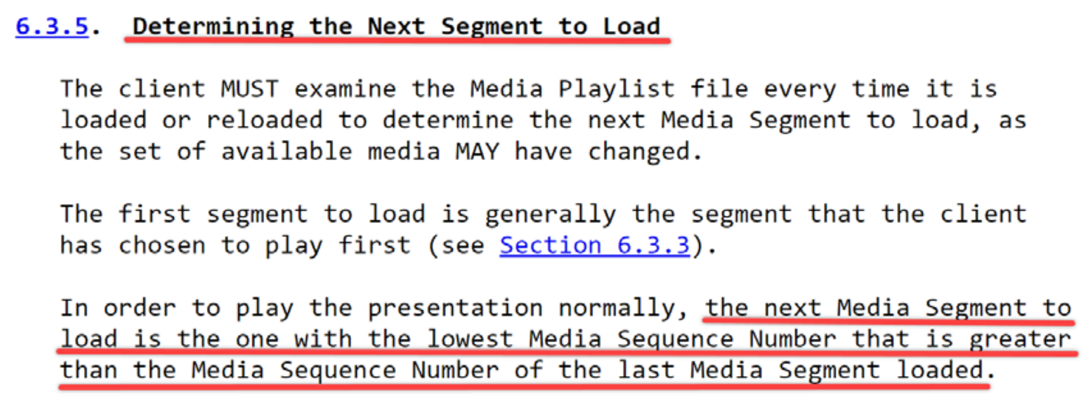
The duration of each Media Segment is indicated in the Media Playlist by its EXTINF tag (Section 4.3.2.1).

Each segment in a Media Playlist has a unique integer Media Sequence Number. The Media Sequence Number of the first segment in the Media Playlist is either 0 or declared in the Playlist (Section 4.3.3.2). The Media Sequence Number of every other segment is equal to the Media Sequence Number of the segment that precedes it plus one.

Each Media Segment MUST carry the continuation of the encoded bitstream from the end of the segment with the previous Media Sequence Number, where values in a series such as timestamps and Continuity Counters MUST continue uninterrupted. The only exceptions are the first Media Segment ever to appear in a Media Playlist and Media Segments that are explicitly signaled as discontinuities (Section 4.3.2.3). Unmarked media discontinuities can trigger playback errors.

Any Media Segment that contains video SHOULD include enough information to initialize a video decoder and decode a continuous set of frames that includes the final frame in the Segment; network efficiency is optimized if there is enough information in the Segment to decode all frames in the Segment. For example, any Media Segment containing H.264 video SHOULD contain an Instantaneous Decoding Refresh (IDR); frames prior to the first IDR will be downloaded but possibly discarded.

https://tools.ietf.org/html/rfc8216



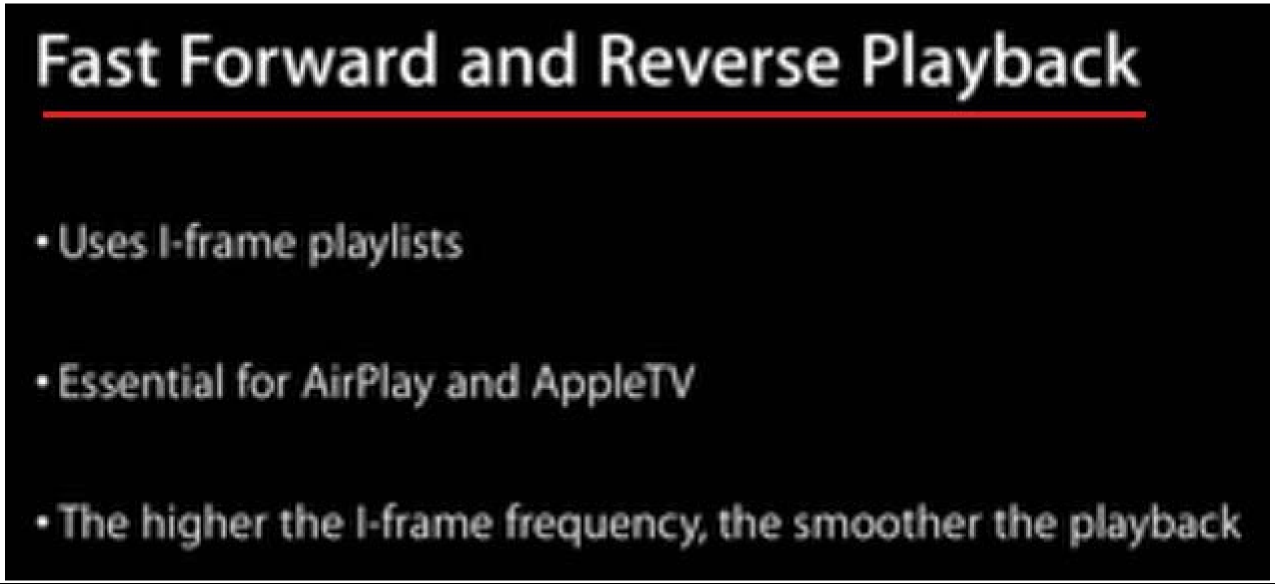
6.3.5. Determining the Next Segment to Load

The client MUST examine the Media Playlist file every time it is loaded or reloaded to determine the next Media Segment to load, as the set of available media MAY have changed.

The first segment to load is generally the segment that the client has chosen to play first (see Section 6.3.3).

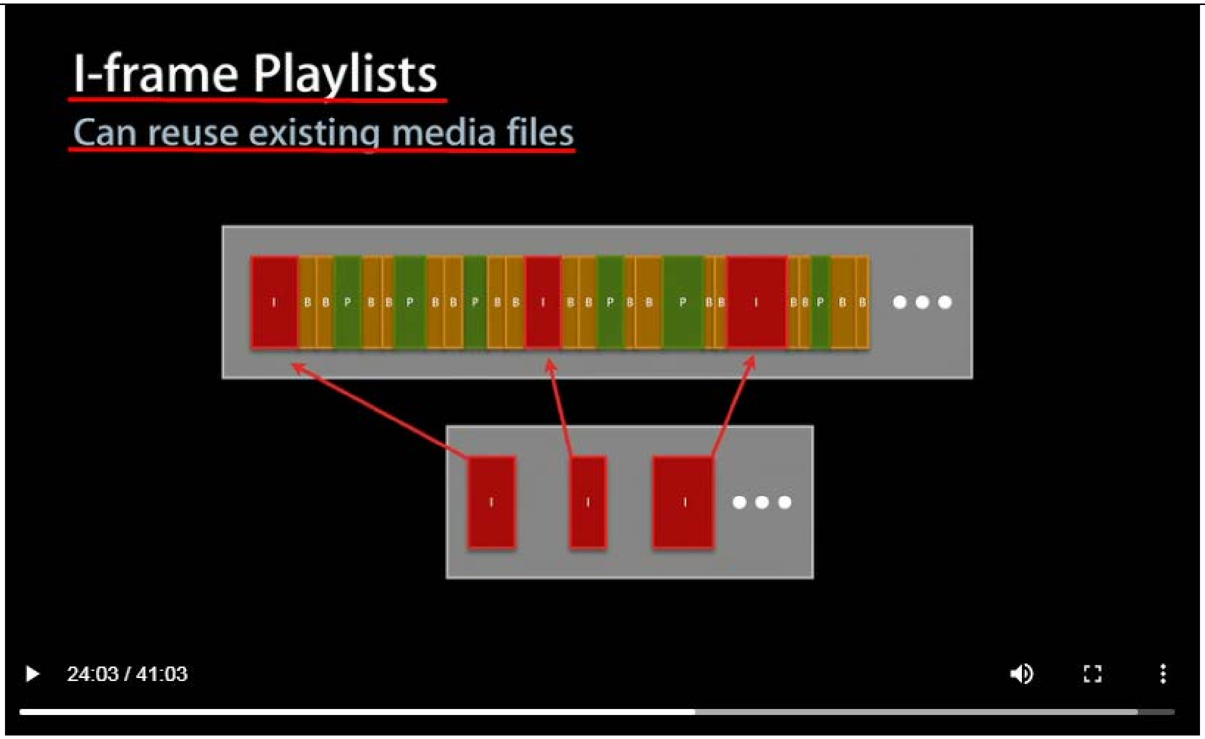
In order to play the presentation normally, the next Media Segment to load is the one with the lowest Media Sequence Number that is greater than the Media Sequence Number of the last Media Segment loaded.

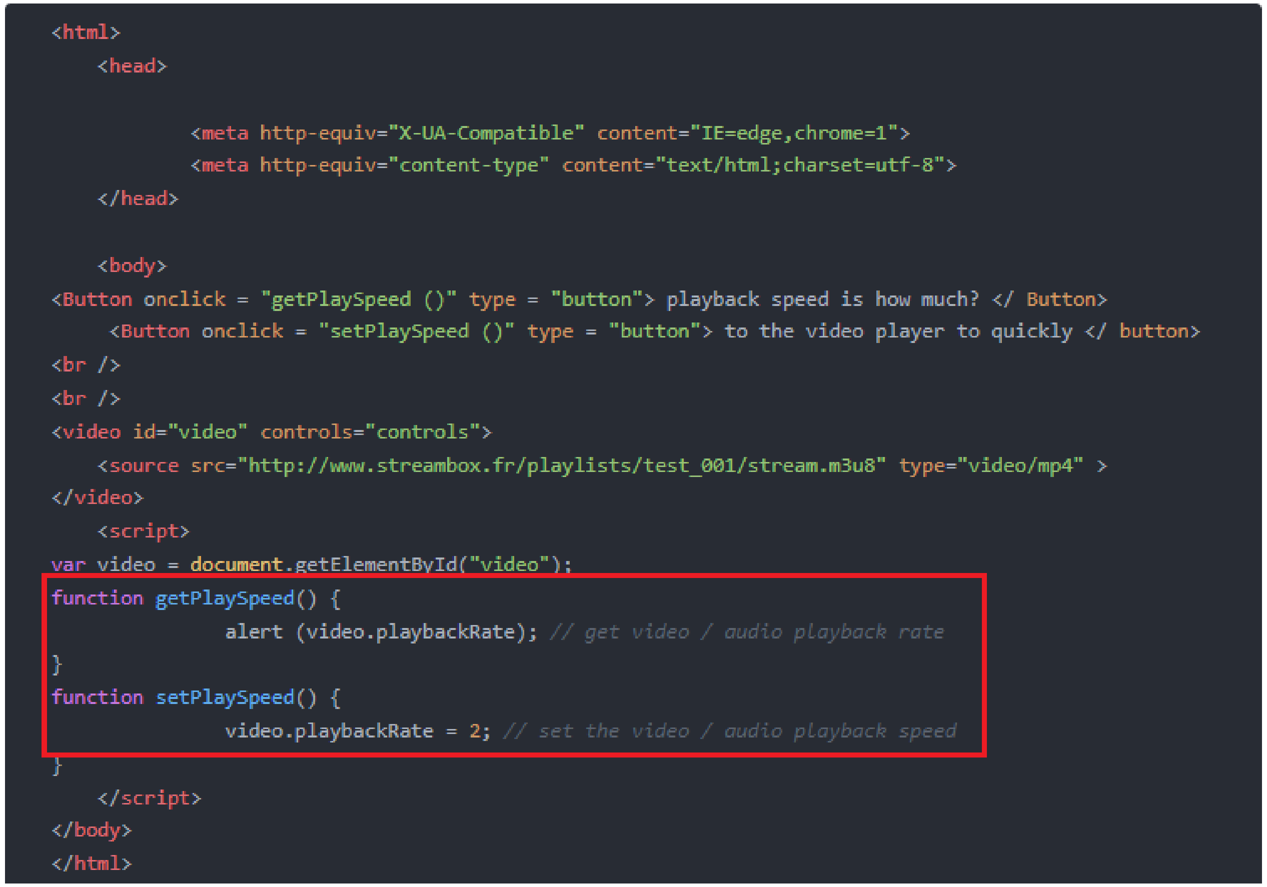
https://tools.ietf.org/html/rfc8216



Fast Forward and Reverse Playback

- Uses I-frame playlists
- Essential for AirPlay and AppleTV
- The higher the I-frame frequency, the smoother the playback

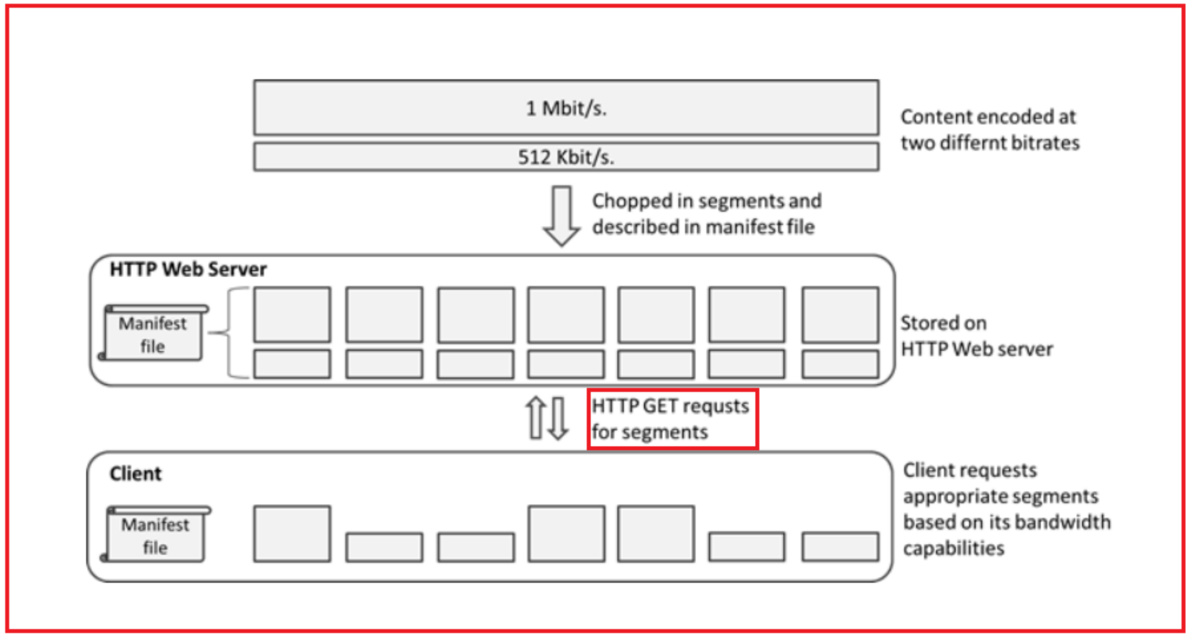
	 I-frame Playlists Can reuse existing media files I B B P B B P B B P B B I B B P B B P B B I B B P B B I I I 24:03 / 41:03 https://developer.apple.com/videos/play/wwdc2012/502/ As shown below, a client sends an HTTP Get request to an HLS media segment distribution server for media segments with video playback at a higher speed than normal speed. Exemplary requests are shown below.

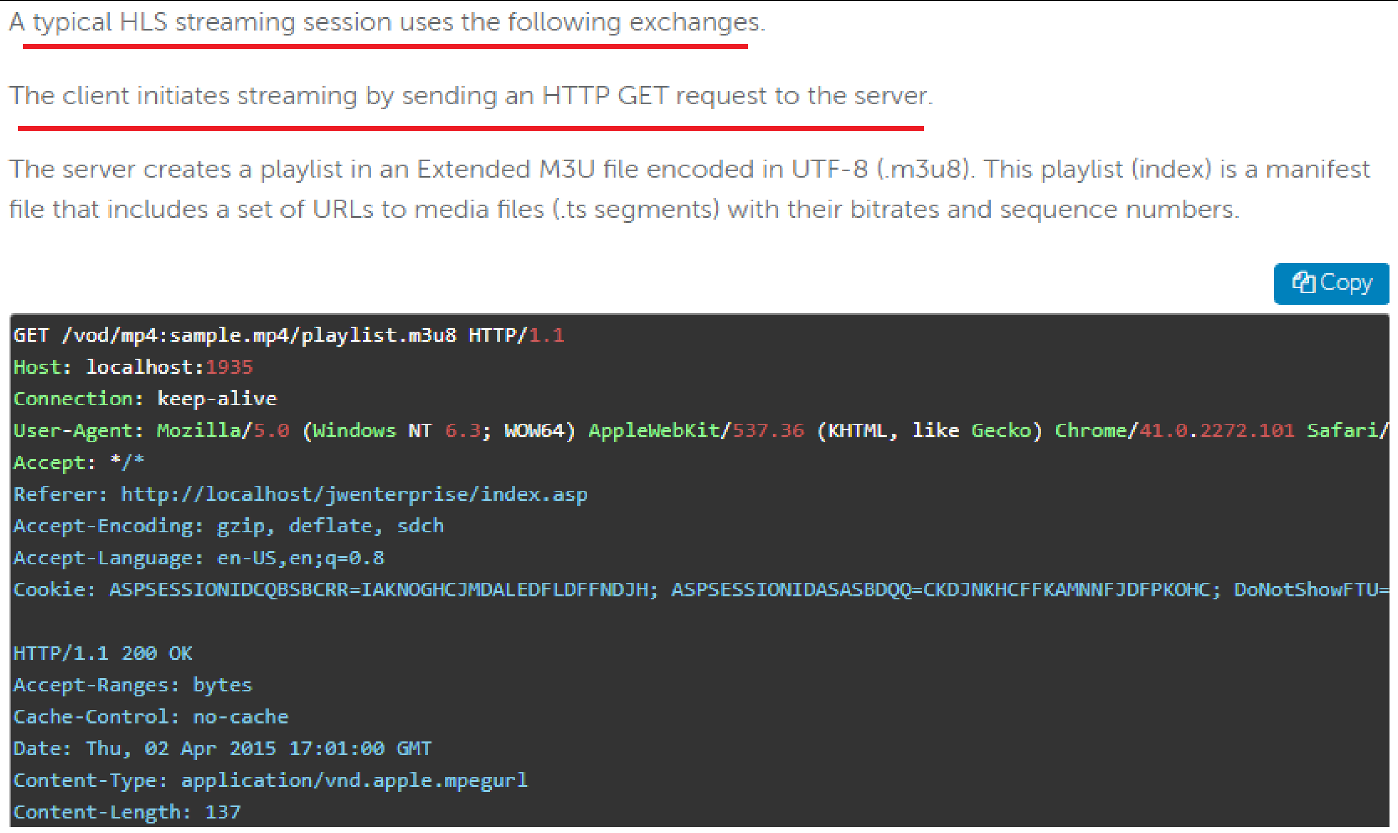
	

```
<html>
    <head>


            <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
            <meta http-equiv="content-type" content="text/html;charset=utf-8">
    </head>

    <body>
<Button onclick = "getPlaySpeed ()" type = "button"> playback speed is how much? </ Button>
     <Button onclick = "setPlaySpeed ()" type = "button"> to the video player to quickly </ button>
<br />
<br />
<video id="video" controls="controls">
    <source src="http://www.streambox.fr/playlists/test_001/stream.m3u8" type="video/mp4" >
</video>
    <script>
var video = document.getElementById("video");
function getPlaySpeed() {
               alert (video.playbackRate); // get video / audio playback rate
}
function setPlaySpeed() {
               video.playbackRate = 2; // set the video / audio playback speed
}
    </script>
</body>
</html>
```

https://programmersought.com/article/32112931107/

transmission of the HTTP GET request to a source appliance; and	The Standard practices transmission of the HTTP GET request (e.g., the HTTP request by the HLS client device) to a source appliance (e.g., HLS server).

	 1 Mbit/s. 512 Kbit/s. Content encoded at two differnt bitrates Chopped in segments and described in manifest file HTTP Web Server Manifest file Stored on HTTP Web server HTTP GET requsts for segments Client Manifest file Client requests appropriate segments based on its bandwidth capabilities http://blog.jpauli.tech/live-video-streaming-stack-from-home/ As shown below, a client sends an HTTP Get request to an HLS media segment distribution server for media segments with URL of the media segments. Exemplary requests are shown below.



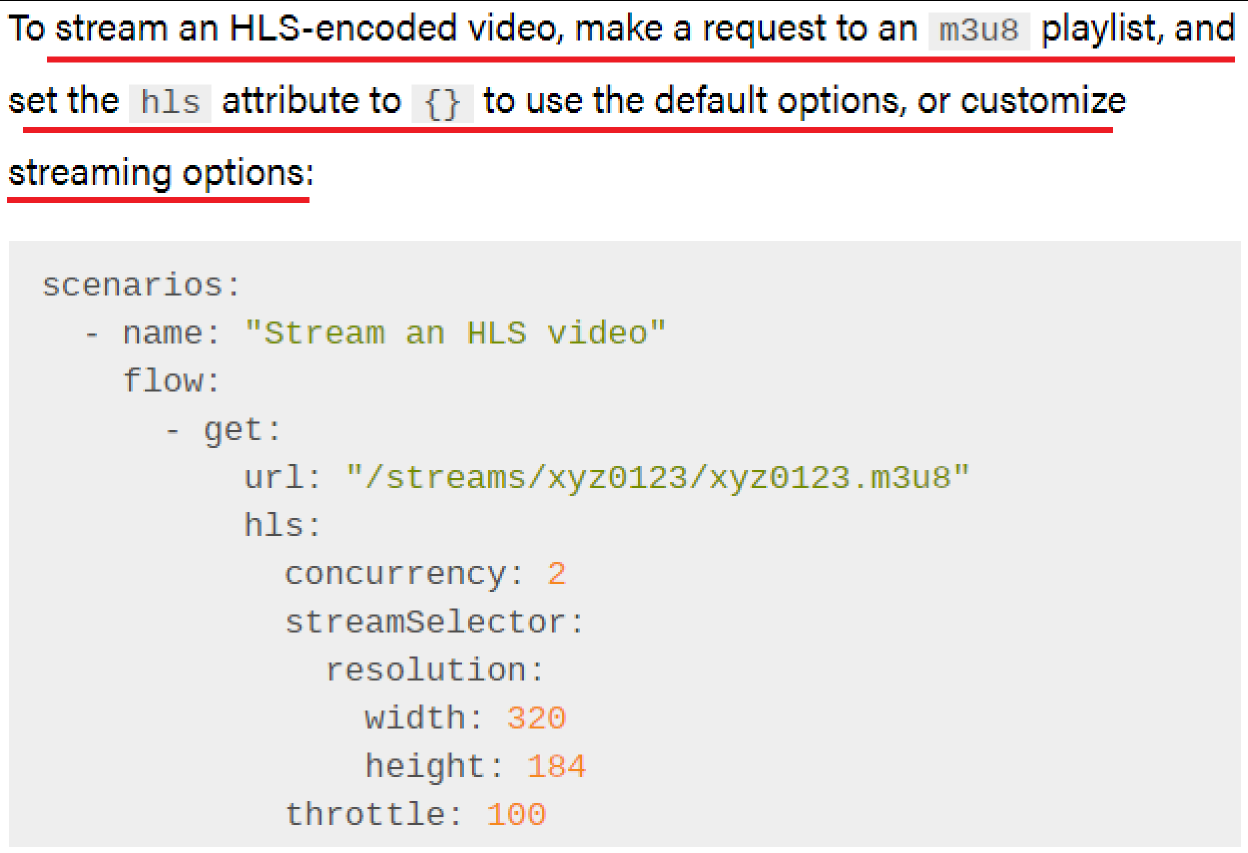
A typical HLS streaming session uses the following exchanges.

The client initiates streaming by sending an HTTP GET request to the server.

The server creates a playlist in an Extended M3U file encoded in UTF-8 (.m3u8). This playlist (index) is a manifest file that includes a set of URLs to media files (.ts segments) with their bitrates and sequence numbers.

```
GET /vod/mp4:sample.mp4/playlist.m3u8 HTTP/1.1
Host: localhost:1935
Connection: keep-alive
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2272.101 Safari/
Accept: */*
Referer: http://localhost/jwenterprise/index.asp
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8
Cookie: ASPSESSIONIDCQBSBCRR=IAKNOGHCJMDALEDFLDFFNDJH; ASPSESSIONIDASASBDQQ=CKDJNKHCFFKAMNNFJDFPKOHC; DoNotShowFTU=

HTTP/1.1 200 OK
Accept-Ranges: bytes
Cache-Control: no-cache
Date: Thu, 02 Apr 2015 17:01:00 GMT
Content-Type: application/vnd.apple.mpegurl
Content-Length: 137
```

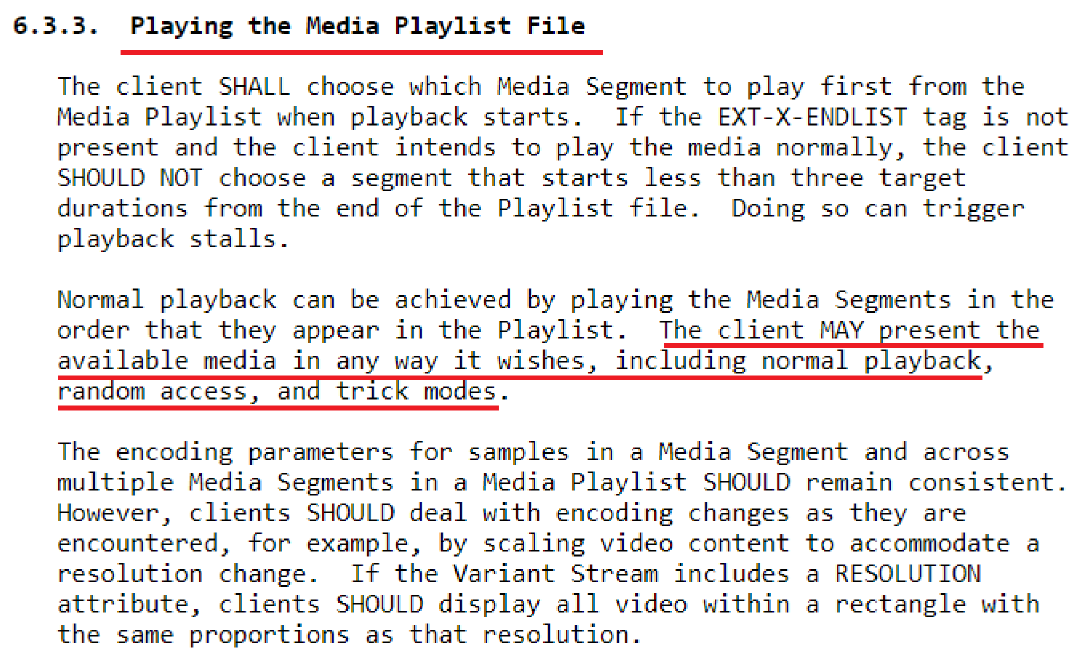
https://www.wowza.com/docs/how-to-troubleshoot-apple-hls-playback

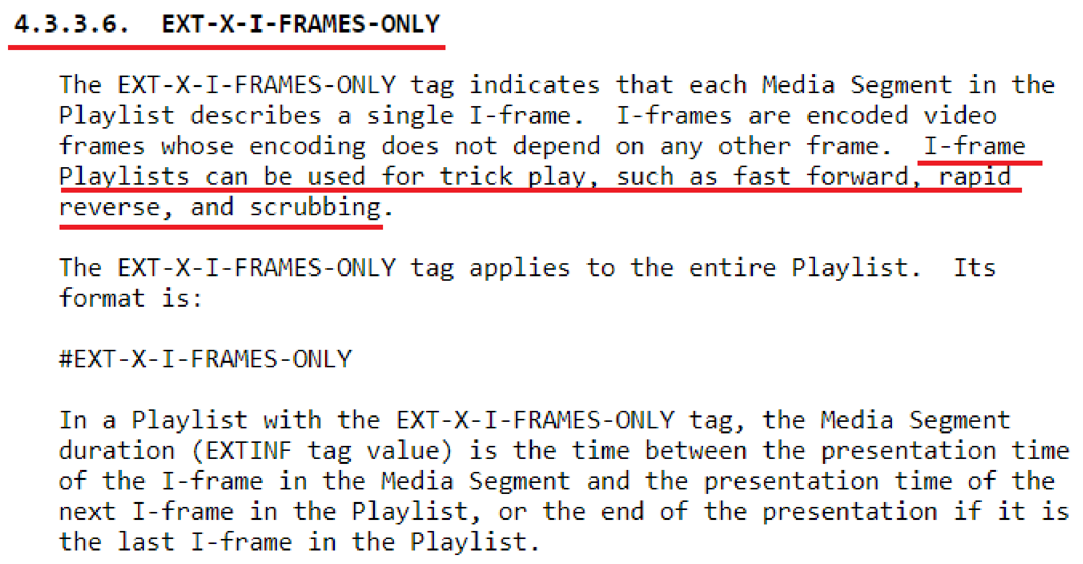


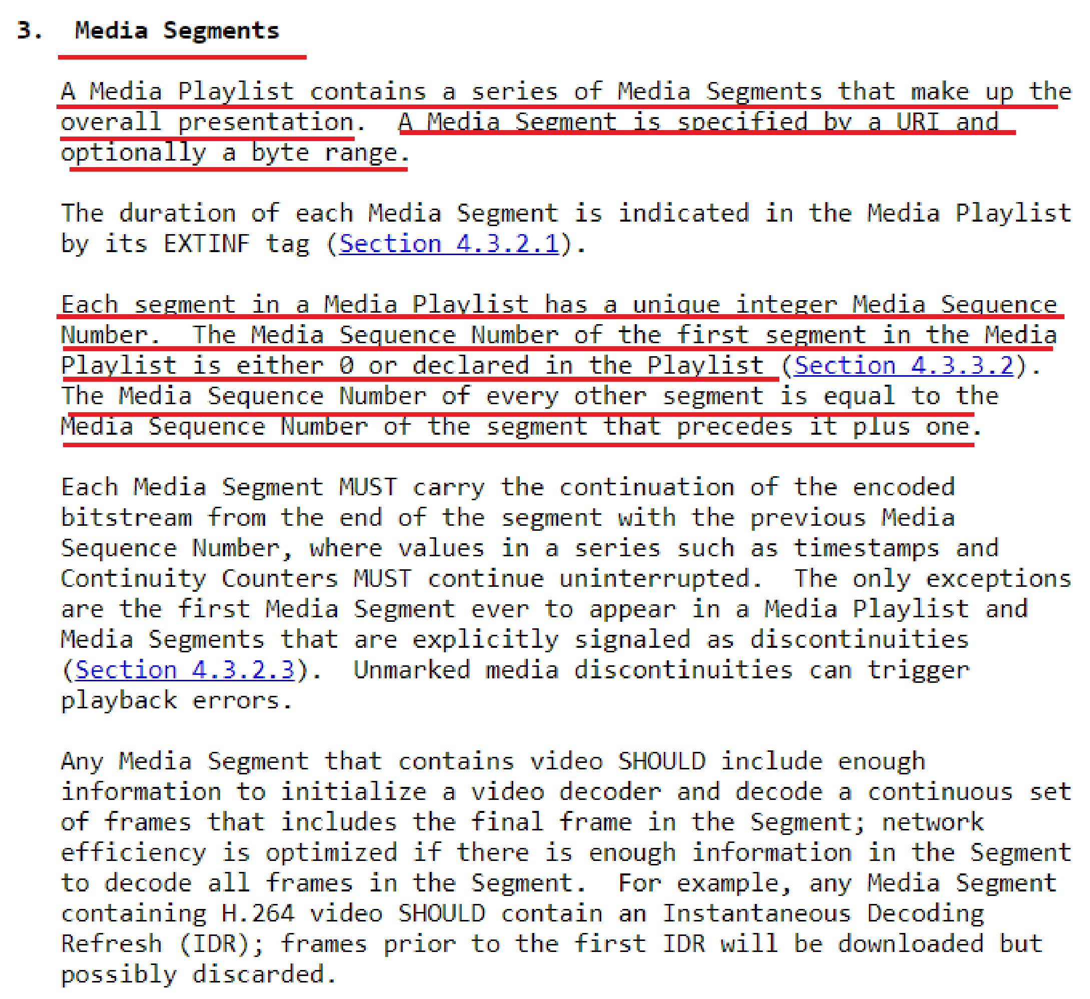
To stream an HLS-encoded video, make a request to an `m3u8` playlist, and set the `hls` attribute to `{}` to use the default options, or customize streaming options:

```
scenarios:
  - name: "Stream an HLS video"
    flow:
      - get:
          url: "/streams/xyz0123/xyz0123.m3u8"
          hls:
            concurrency: 2
            streamSelector:
              resolution:
                width: 320
                height: 184
            throttle: 100
```

https://artillery.io/docs/guides/plugins/plugin-hls.html

	 6.3.3. Playing the Media Playlist File The client SHALL choose which Media Segment to play first from the Media Playlist when playback starts. If the EXT-X-ENDLIST tag is not present and the client intends to play the media normally, the client SHOULD NOT choose a segment that starts less than three target durations from the end of the Playlist file. Doing so can trigger playback stalls. Normal playback can be achieved by playing the Media Segments in the order that they appear in the Playlist. The client MAY present the available media in any way it wishes, including normal playback, random access, and trick modes. The encoding parameters for samples in a Media Segment and across multiple Media Segments in a Media Playlist SHOULD remain consistent. However, clients SHOULD deal with encoding changes as they are encountered, for example, by scaling video content to accommodate a resolution change. If the Variant Stream includes a RESOLUTION attribute, clients SHOULD display all video within a rectangle with the same proportions as that resolution. https://tools.ietf.org/html/rfc8216

	 **4.3.3.6. EXT-X-I-FRAMES-ONLY** The EXT-X-I-FRAMES-ONLY tag indicates that each Media Segment in the Playlist describes a single I-frame. I-frames are encoded video frames whose encoding does not depend on any other frame. I-frame Playlists can be used for trick play, such as fast forward, rapid reverse, and scrubbing. The EXT-X-I-FRAMES-ONLY tag applies to the entire Playlist. Its format is: #EXT-X-I-FRAMES-ONLY In a Playlist with the EXT-X-I-FRAMES-ONLY tag, the Media Segment duration (EXTINF tag value) is the time between the presentation time of the I-frame in the Media Segment and the presentation time of the next I-frame in the Playlist, or the end of the presentation if it is the last I-frame in the Playlist. https://tools.ietf.org/html/rfc8216



3. Media Segments

A Media Playlist contains a series of Media Segments that make up the overall presentation. A Media Segment is specified by a URI and optionally a byte range.

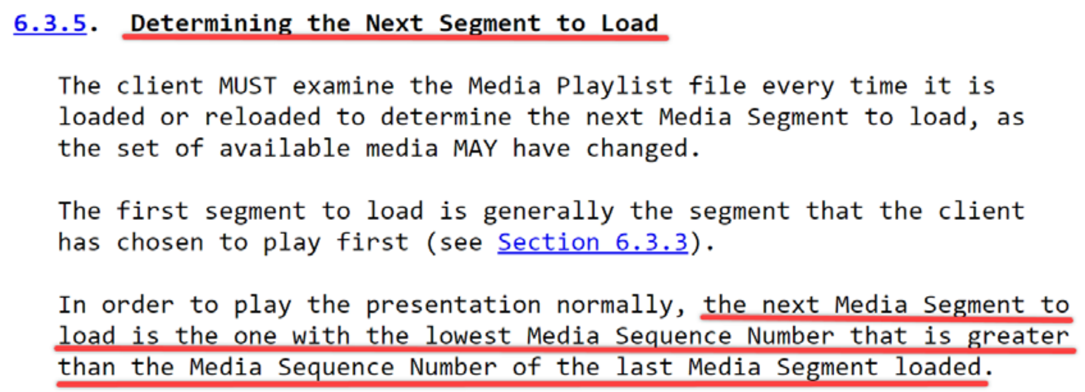
The duration of each Media Segment is indicated in the Media Playlist by its EXTINF tag (Section 4.3.2.1).

Each segment in a Media Playlist has a unique integer Media Sequence Number. The Media Sequence Number of the first segment in the Media Playlist is either 0 or declared in the Playlist (Section 4.3.3.2). The Media Sequence Number of every other segment is equal to the Media Sequence Number of the segment that precedes it plus one.

Each Media Segment MUST carry the continuation of the encoded bitstream from the end of the segment with the previous Media Sequence Number, where values in a series such as timestamps and Continuity Counters MUST continue uninterrupted. The only exceptions are the first Media Segment ever to appear in a Media Playlist and Media Segments that are explicitly signaled as discontinuities (Section 4.3.2.3). Unmarked media discontinuities can trigger playback errors.

Any Media Segment that contains video SHOULD include enough information to initialize a video decoder and decode a continuous set of frames that includes the final frame in the Segment; network efficiency is optimized if there is enough information in the Segment to decode all frames in the Segment. For example, any Media Segment containing H.264 video SHOULD contain an Instantaneous Decoding Refresh (IDR); frames prior to the first IDR will be downloaded but possibly discarded.

https://tools.ietf.org/html/rfc8216



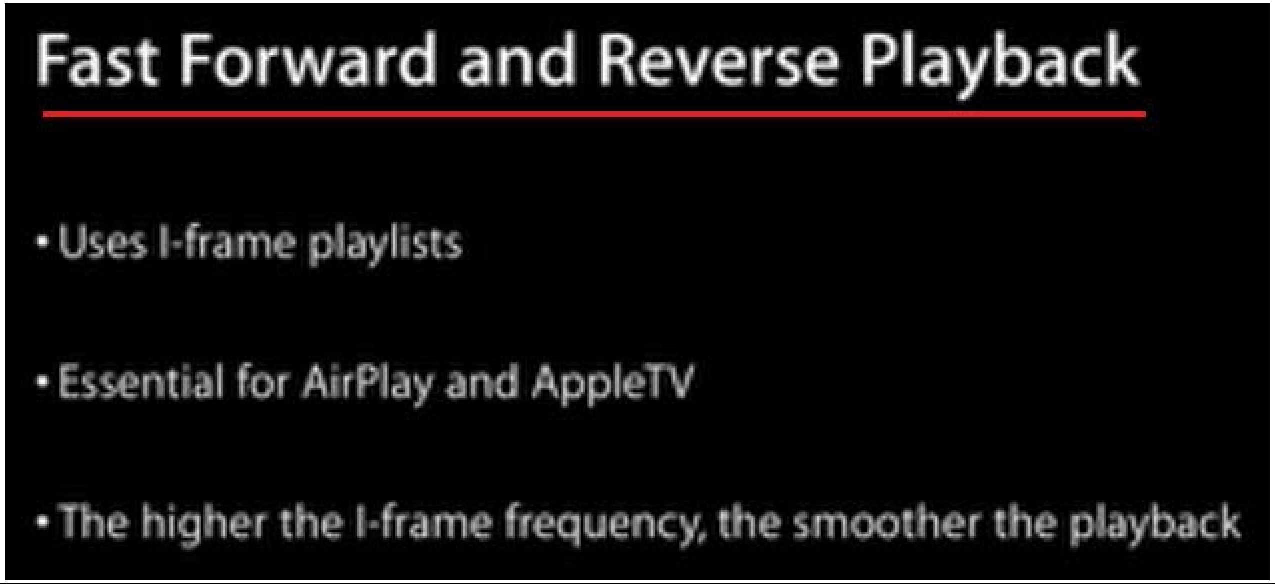
6.3.5. Determining the Next Segment to Load

The client MUST examine the Media Playlist file every time it is loaded or reloaded to determine the next Media Segment to load, as the set of available media MAY have changed.

The first segment to load is generally the segment that the client has chosen to play first (see Section 6.3.3).

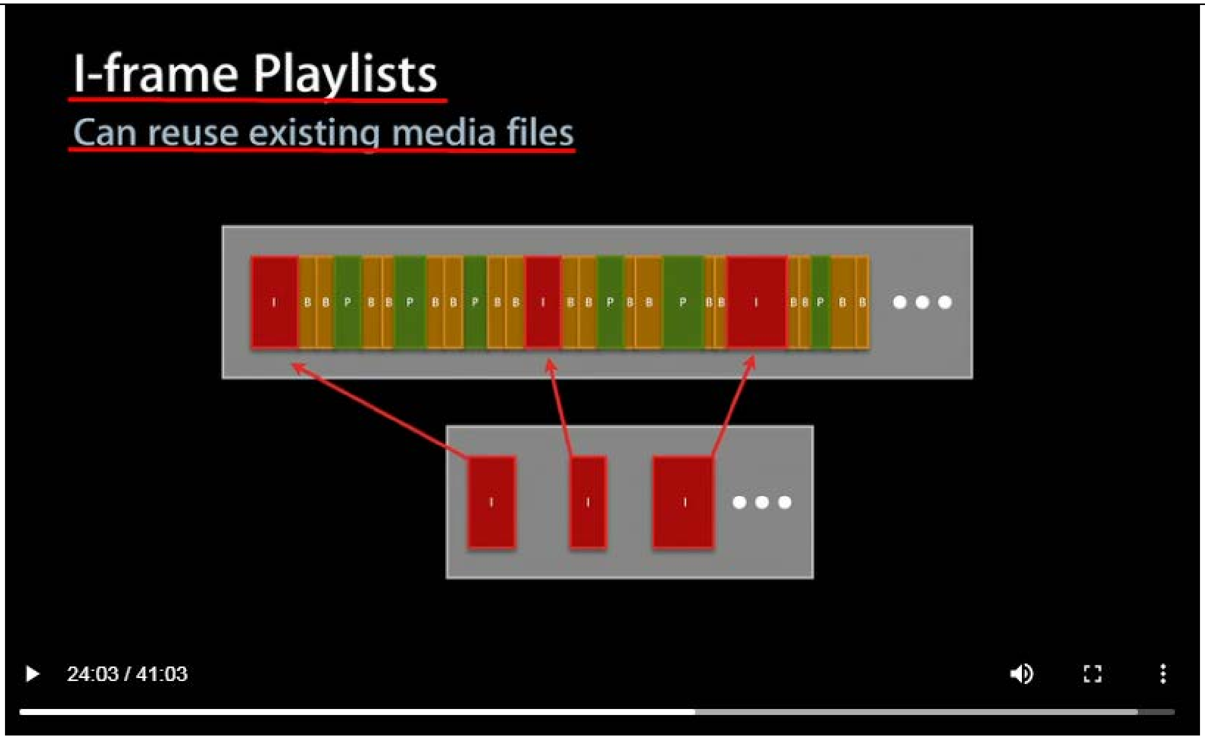
In order to play the presentation normally, the next Media Segment to load is the one with the lowest Media Sequence Number that is greater than the Media Sequence Number of the last Media Segment loaded.

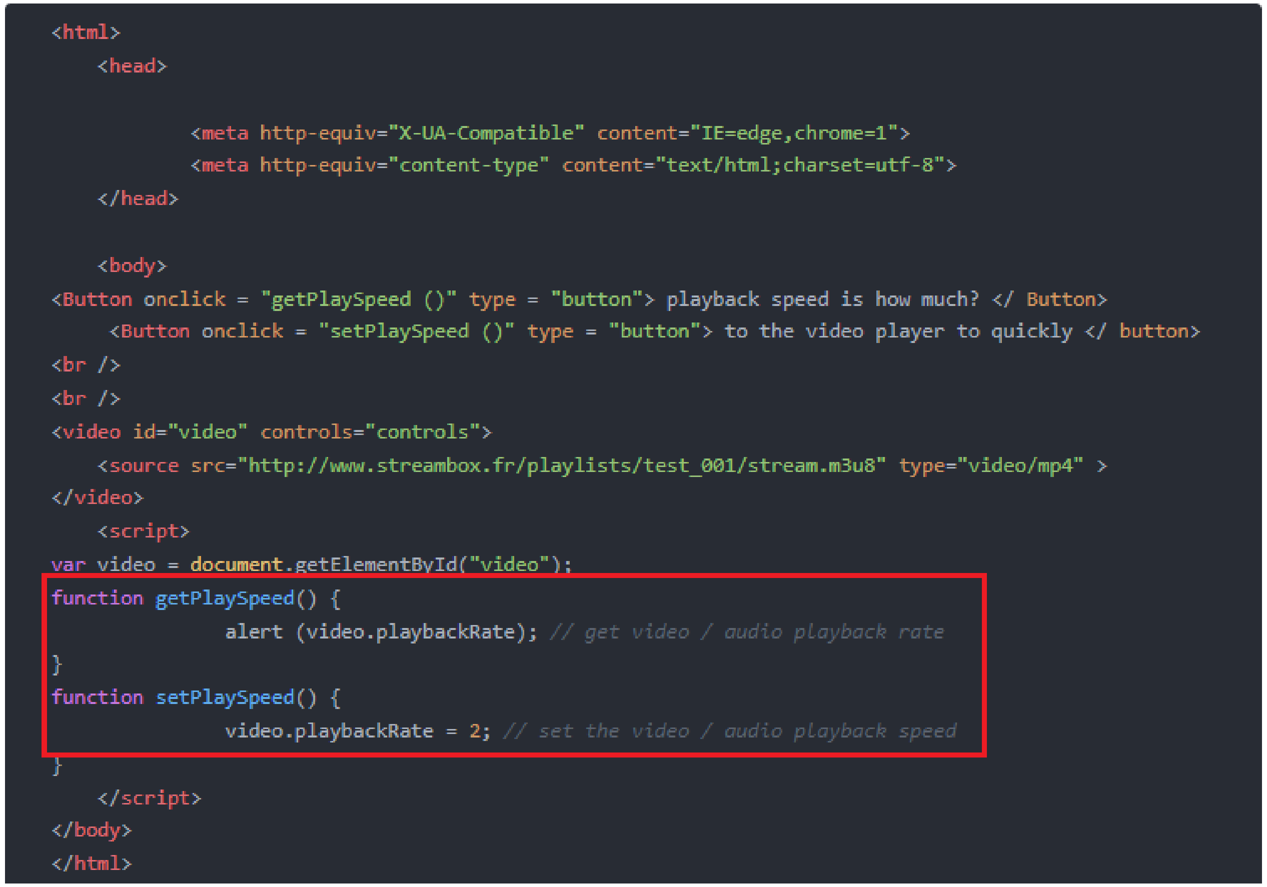
https://tools.ietf.org/html/rfc8216



Fast Forward and Reverse Playback

- Uses I-frame playlists
- Essential for AirPlay and AppleTV
- The higher the I-frame frequency, the smoother the playback

	 I-frame Playlists Can reuse existing media files I B B P B B P B B P B B I B B P B B P B B I B B P B B I I I 24:03 / 41:03 https://developer.apple.com/videos/play/wwdc2012/502/ As shown below, a client sends an HTTP Get request to an HLS media segment distribution server for media segments with video playback at a higher speed than normal speed. Exemplary requests are shown below.

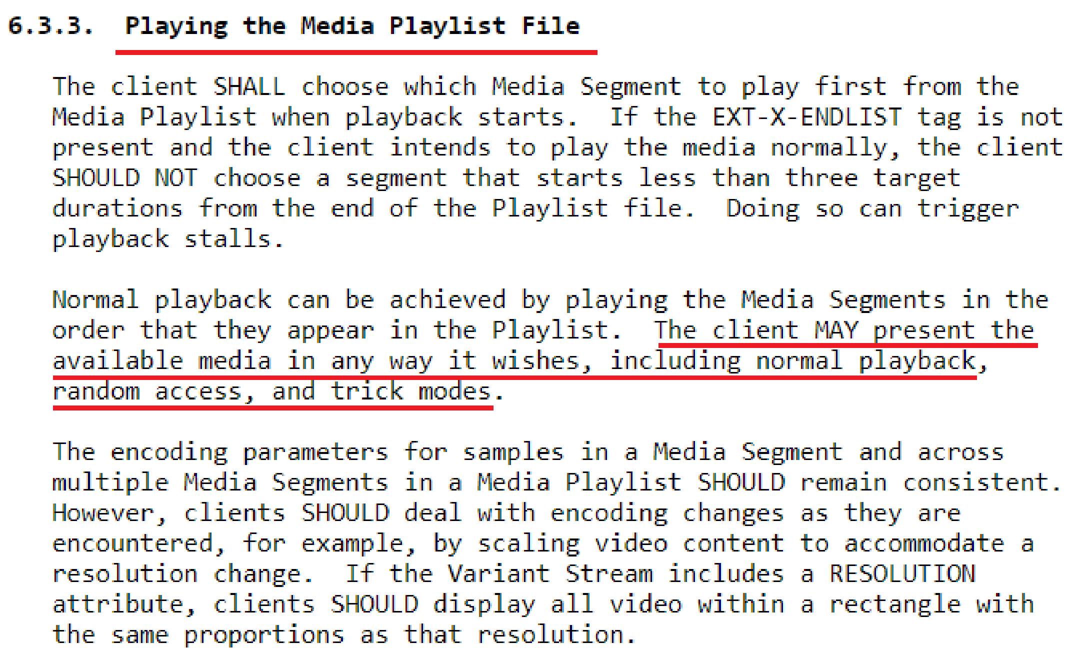


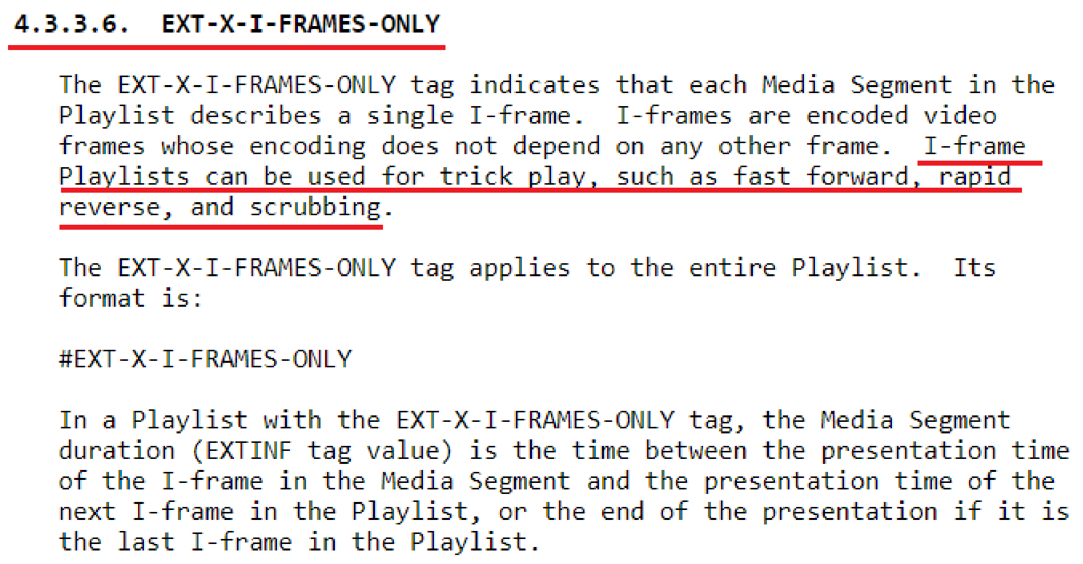
```
<html>
    <head>


            <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
            <meta http-equiv="content-type" content="text/html;charset=utf-8">
    </head>

    <body>
<Button onclick = "getPlaySpeed ()" type = "button"> playback speed is how much? </ Button>
     <Button onclick = "setPlaySpeed ()" type = "button"> to the video player to quickly </ button>
<br />
<br />
<video id="video" controls="controls">
    <source src="http://www.streambox.fr/playlists/test_001/stream.m3u8" type="video/mp4" >
</video>
    <script>
var video = document.getElementById("video");
function getPlaySpeed() {
               alert (video.playbackRate); // get video / audio playback rate
}
function setPlaySpeed() {
               video.playbackRate = 2; // set the video / audio playback speed
}
    </script>
</body>
</html>
```

https://programmersought.com/article/32112931107/

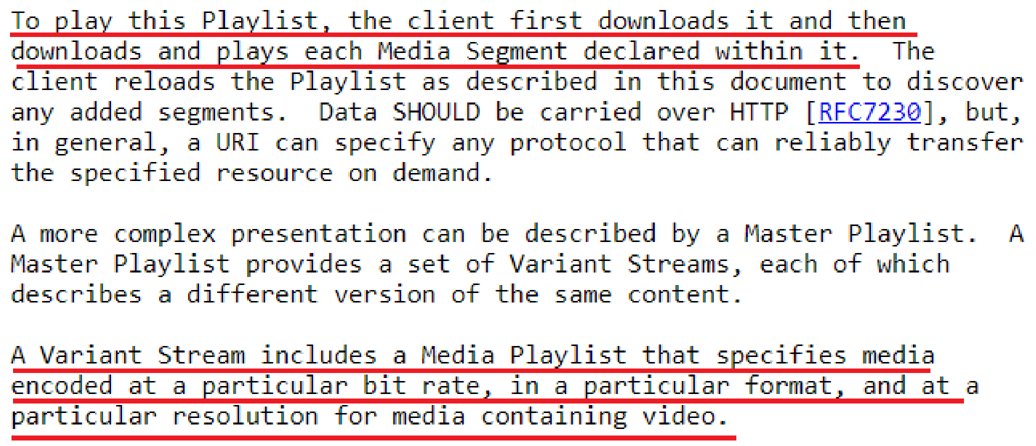
discontinuous transmission, in sections, of selected video frames of an original encoded video stream from the source	The Standard practices discontinuous transmission (e.g., discontinues transmission of media segments during a trick mode operation such as fast-forward, fast-rewind, etc.), in sections (e.g., media segments), of selected video frames (e.g., selected video frames in trick mode which skips other frames and only plays I frames) of an original encoded video stream from the source appliance (e.g., HTTP server) to a destination appliance (e.g., HLS client)

appliance to a destination appliance in a HTTP response using an extended HTTP chunked transfer encoding mode, in which the selected encoded video frames for the fast search operation are transported in respective chunks, wherein each chunk includes one complete respective selected encoded video frame in a second part and information about a starting time, as located in the original encoded video stream, of the respective selected video frame in a first part, wherein the second part is different from the first part and the information about a starting time of the respective selected video frame being positioned in a commentary line of the first part.	in a HTTP response using an extended HTTP chunked transfer encoding mode (e.g., transferring video data in segments or chunks in encoding format), in which the selected encoded video frames for the fast search operation (e.g., trick mode operation such as fast forward or rewind) are transported in respective chunks (e.g., segments or chunks), wherein each chunk includes one complete respective selected encoded video frame (e.g., a segment that comprises an I-frame) in a second part (e.g., the media segment) and information about a starting time (e.g., media segment timestamp), as located in the original encoded video stream (media segments for trick mode are time-aligned with the segments in the original media streams, e.g., the original video stream, which enables switching between the trick mode stream and the normal play rate streams), of the respective selected video frame (e.g., the selected video data frame in trick mode) in a first part (e.g., media segment details such as media segment timestamp, sequence no., etc.), wherein the second part (e.g., the media segment) is different from the first part (e.g., media segment details such as media segment timestamp, sequence no., etc.) and the information about a starting time of the respective selected video frame being positioned in a commentary line of the first part (e.g., media segment details such as media segment timestamp, sequence no., etc.).

	 **6.3.3. Playing the Media Playlist File** The client SHALL choose which Media Segment to play first from the Media Playlist when playback starts. If the EXT-X-ENDLIST tag is not present and the client intends to play the media normally, the client SHOULD NOT choose a segment that starts less than three target durations from the end of the Playlist file. Doing so can trigger playback stalls. Normal playback can be achieved by playing the Media Segments in the order that they appear in the Playlist. The client MAY present the available media in any way it wishes, including normal playback, random access, and trick modes. The encoding parameters for samples in a Media Segment and across multiple Media Segments in a Media Playlist SHOULD remain consistent. However, clients SHOULD deal with encoding changes as they are encountered, for example, by scaling video content to accommodate a resolution change. If the Variant Stream includes a RESOLUTION attribute, clients SHOULD display all video within a rectangle with the same proportions as that resolution. https://tools.ietf.org/html/rfc8216

	 **4.3.3.6. EXT-X-I-FRAMES-ONLY** The EXT-X-I-FRAMES-ONLY tag indicates that each Media Segment in the Playlist describes a single I-frame. I-frames are encoded video frames whose encoding does not depend on any other frame. I-frame Playlists can be used for trick play, such as fast forward, rapid reverse, and scrubbing. The EXT-X-I-FRAMES-ONLY tag applies to the entire Playlist. Its format is: `#EXT-X-I-FRAMES-ONLY` In a Playlist with the EXT-X-I-FRAMES-ONLY tag, the Media Segment duration (EXTINF tag value) is the time between the presentation time of the I-frame in the Media Segment and the presentation time of the next I-frame in the Playlist, or the end of the presentation if it is the last I-frame in the Playlist. https://tools.ietf.org/html/rfc8216

	Media Segments A Media Playlist contains a series of Media Segments that make up the overall presentation. A Media Segment is specified by a URI and optionally a byte range. The duration of each Media Segment is indicated in the Media Playlist by its EXTINF tag (Section 4.3.2.1). Each segment in a Media Playlist has a unique integer Media Sequence Number. The Media Sequence Number of the first segment in the Media Playlist is either 0 or declared in the Playlist (Section 4.3.3.2). The Media Sequence Number of every other segment is equal to the Media Sequence Number of the segment that precedes it plus one. Each Media Segment MUST carry the continuation of the encoded bitstream from the end of the segment with the previous Media Sequence Number, where values in a series such as timestamps and Continuity Counters MUST continue uninterrupted. The only exceptions are the first Media Segment ever to appear in a Media Playlist and Media Segments that are explicitly signaled as discontinuities (Section 4.3.2.3). Unmarked media discontinuities can trigger playback errors. Any Media Segment that contains video SHOULD include enough information to initialize a video decoder and decode a continuous set of frames that includes the final frame in the Segment; network efficiency is optimized if there is enough information in the Segment to decode all frames in the Segment. For example, any Media Segment containing H.264 video SHOULD contain an Instantaneous Decoding Refresh (IDR); frames prior to the first IDR will be downloaded but possibly discarded. https://tools.ietf.org/html/rfc8216

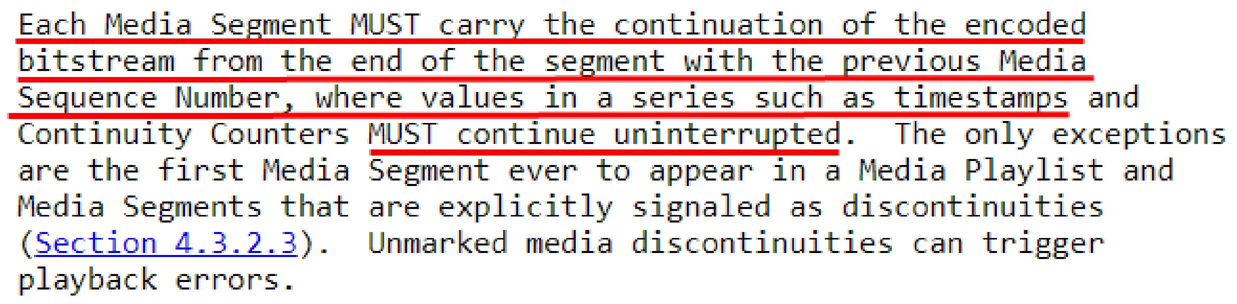


To play this Playlist, the client first downloads it and then downloads and plays each Media Segment declared within it. The client reloads the Playlist as described in this document to discover any added segments. Data SHOULD be carried over HTTP [RFC7230], but, in general, a URI can specify any protocol that can reliably transfer the specified resource on demand.

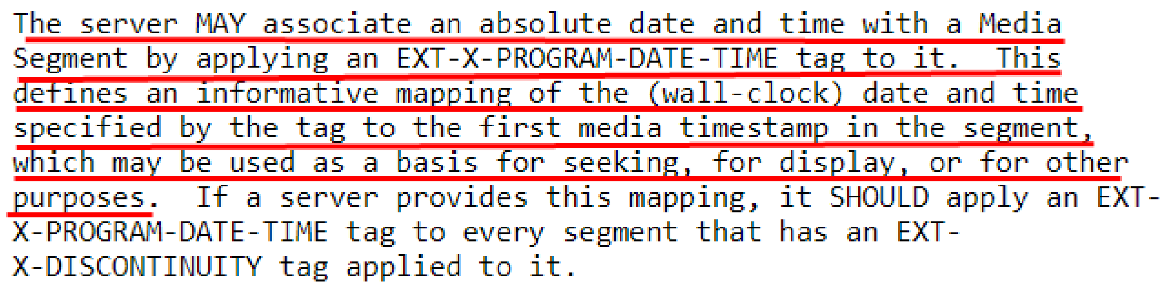
A more complex presentation can be described by a Master Playlist. A Master Playlist provides a set of Variant Streams, each of which describes a different version of the same content.

A Variant Stream includes a Media Playlist that specifies media encoded at a particular bit rate, in a particular format, and at a particular resolution for media containing video.

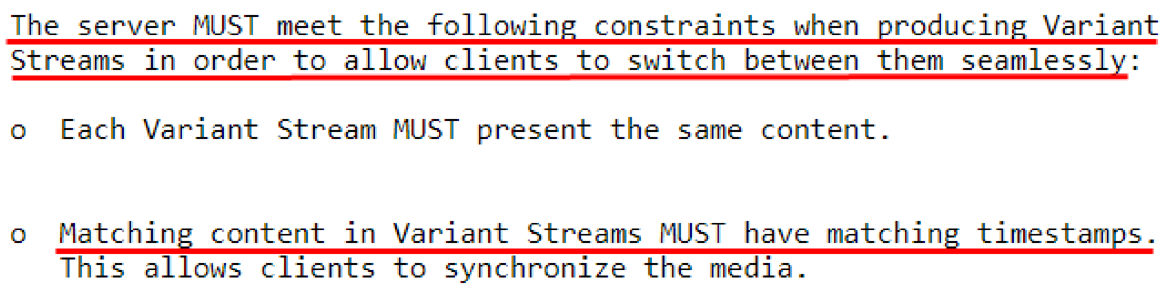
https://tools.ietf.org/html/rfc8216



Each Media Segment MUST carry the continuation of the encoded bitstream from the end of the segment with the previous Media Sequence Number, where values in a series such as timestamps and Continuity Counters MUST continue uninterrupted. The only exceptions are the first Media Segment ever to appear in a Media Playlist and Media Segments that are explicitly signaled as discontinuities (Section 4.3.2.3). Unmarked media discontinuities can trigger playback errors.



The server MAY associate an absolute date and time with a Media Segment by applying an EXT-X-PROGRAM-DATE-TIME tag to it. This defines an informative mapping of the (wall-clock) date and time specified by the tag to the first media timestamp in the segment, which may be used as a basis for seeking, for display, or for other purposes. If a server provides this mapping, it SHOULD apply an EXT-X-PROGRAM-DATE-TIME tag to every segment that has an EXT-X-DISCONTINUITY tag applied to it.

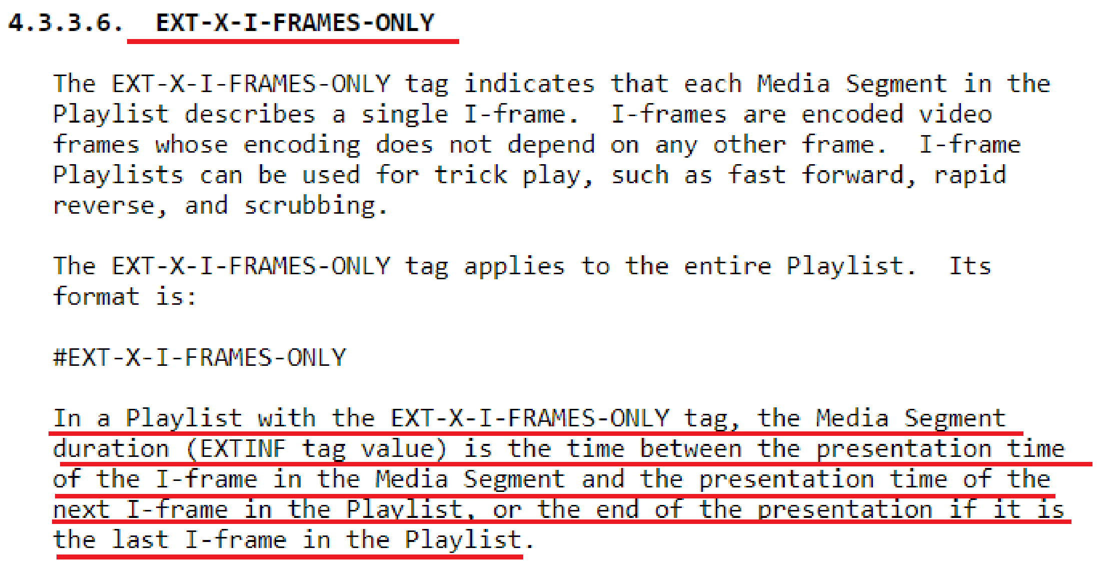


The server MUST meet the following constraints when producing Variant Streams in order to allow clients to switch between them seamlessly:

o Each Variant Stream MUST present the same content.

o Matching content in Variant Streams MUST have matching timestamps. This allows clients to synchronize the media.

https://tools.ietf.org/html/rfc8216



4.3.3.6. EXT-X-I-FRAMES-ONLY

The EXT-X-I-FRAMES-ONLY tag indicates that each Media Segment in the Playlist describes a single I-frame. I-frames are encoded video frames whose encoding does not depend on any other frame. I-frame Playlists can be used for trick play, such as fast forward, rapid reverse, and scrubbing.

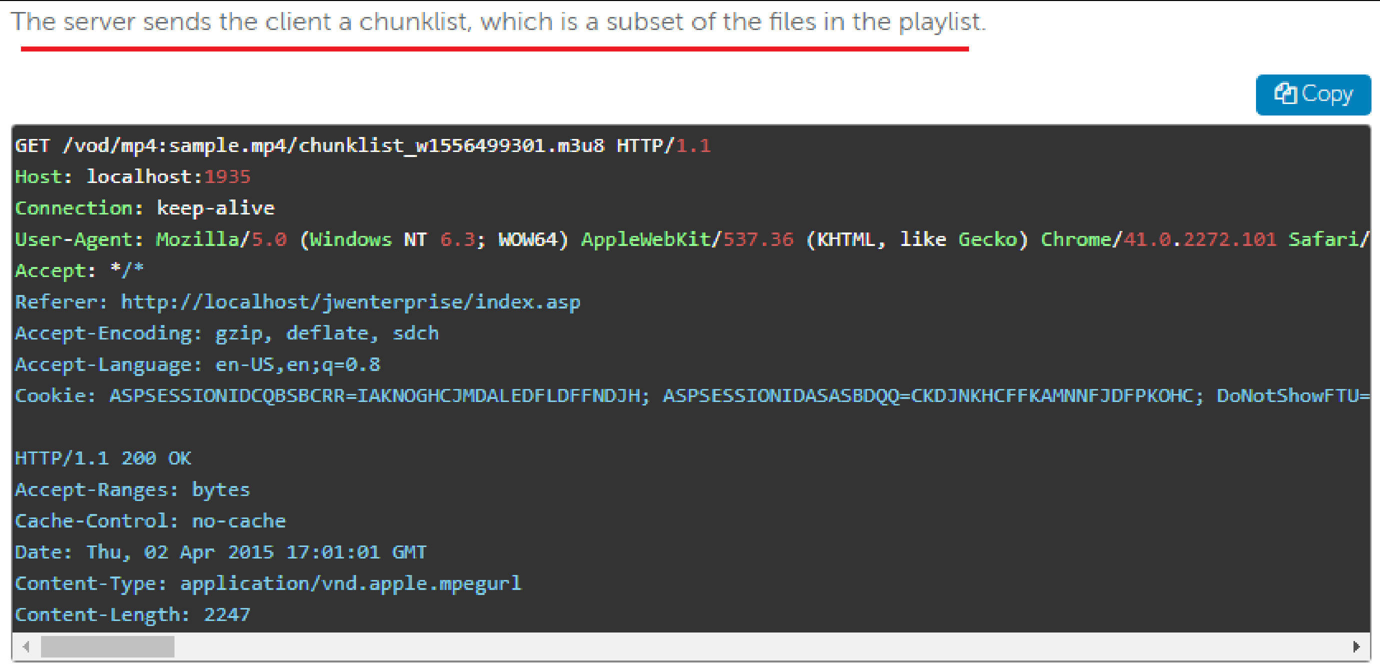
The EXT-X-I-FRAMES-ONLY tag applies to the entire Playlist. Its format is:

```
#EXT-X-I-FRAMES-ONLY
```

In a Playlist with the EXT-X-I-FRAMES-ONLY tag, the Media Segment duration (EXTINF tag value) is the time between the presentation time of the I-frame in the Media Segment and the presentation time of the next I-frame in the Playlist, or the end of the presentation if it is the last I-frame in the Playlist.

https://tools.ietf.org/html/rfc8216#section-4.3.3.6

Exemplary server response is shown below.



The server sends the client a chunklist, which is a subset of the files in the playlist.

```
GET /vod/mp4:sample.mp4/chunklist_w1556499301.m3u8 HTTP/1.1
Host: localhost:1935
Connection: keep-alive
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2272.101 Safari/
Accept: */*
Referer: http://localhost/jwenterprise/index.asp
Accept-Encoding: gzip, deflate, sdch
Accept-Language: en-US,en;q=0.8
Cookie: ASPSESSIONIDCQBSBCRR=IAKNOGHCJMDALEDFLDFFNDJH; ASPSESSIONIDASASBDQQ=CKDJNKHCFFKAMNNFJDFPKOHC; DoNotShowFTU=

HTTP/1.1 200 OK
Accept-Ranges: bytes
Cache-Control: no-cache
Date: Thu, 02 Apr 2015 17:01:01 GMT
Content-Type: application/vnd.apple.mpegurl
Content-Length: 2247
```

https://www.wowza.com/docs/how-to-troubleshoot-apple-hls-playback